



SISTEMA ASISTENCIA

Documentation



OCTOBER 5, 2019
DEPARTMENT OF COMPUTER SCIENCE
Atmiya University

Table of Contents

Project Profile	8
SPECIFICATION	9
Project Directory Structure.....	11
SYSTEM ANALYSIS	12
Existing System.....	12
Drawbacks of existing system:	12
Requirements of New System	12
Advantage Of New System	12
User requirement:.....	13
Feasibility Study:	13
Use case	16
What is a use case diagram?	16
When to apply use case diagrams?	16
UML use case diagrams are ideal for:.....	17
Use case diagram components	17
Use case diagram symbols and notation	17
System	18
Use Case:	18
Actors:	18
Relationships:	18
Use case diagram for Attendance system	19
Drawing Tool Used- : Draw.io	20
NodeJS	21
What is Node.js?.....	21
Why Node.js?.....	21
What Can Node.js Do?.....	21
What is a Node.js File?	22
Node Package Manager	23
What is NPM?.....	23

More on NPM?	23
What is a Package?	23
Use NPM to	23
Install Node.js & NPM on Windows	24
USER INTERFACE	28
What is User Interface Design?	28
There are 3 issues / parts of a user interface design, these being.....	28
Key Components of UI/UX Design.....	29
Interaction Design	30
Usability	30
Wire framing.....	30
Visual Design.....	30
Screenshots.....	32
Data Flow Diagram – DFD	43
Physical and logical data flow diagrams	43
Symbols and Notations Used in DFDs	43
Data flow diagram Sistema Asistencia	46
Data Dictionary	49
Teachers:	50
Students:	50
Attendance:.....	51
PUG.....	52
What is Pug Jade.....	52
Pug.js to make your life easier with HTML templates.....	52
Let's see how pug code looks like.....	53
PUG Structure index.pug vs. index.html.....	53
Frameworks & Libraries.....	54
Express js.....	54
TypeORM	54
JWT	54

FaceAPI.....	55
TypeScript	55

ACKNOWLEDGEMENT

Sistema Assentensia | NodeJS

Date:

To Whom It May Concern:

I would like to highlight the fact that this project is the product of many contributors. First, I'd like to thank Professor **Chiraj Jagani** who accepted to work with me on my Project idea, supervising my work.

Second, I would like to thank the **Google, Youtube, stack overflow** community for providing tremendous help to the elaboration of this project.

ABSTRACT

Sistema Assentensia | NodeJS

Student attendance management system deals with the maintenance of the student's attendance details. It generates the attendance of the student on basis of presence in class. It is maintained on the daily basis of their attendance. The staffs will be provided with the separate username & password to make the student's status. The staffs handling the particular subjects responsible to make the attendance for all students. Only if the student present on that particular period, then the attendance will be calculated. The students attendance reports based on weekly and consolidate will be generated.

INTRODUCTION

Sistema Assentensia

Overview

- ✓ THE ONLINE ATTENDANCE SYSTEM focuses on creating effective assessment of the applicant. This project will enable educational institutes to take an attendance and add teachers and students. This is specially designed for conducting digital Attendance
- ✓ Here principal can add faculties and students and can see the attendance taken by the teachers, in the filtering forms.
- ✓ Here, faculty have rights to add students and can also take attendance by capturing or adding a photograph and can see the attendance taken by them , in the filtering forms.
- ✓ It can be conducted for remote attendance and can take attendance fully automated. This site is an attempt to remove flaws of existing forms or patterns of taking attendances

INTRODUCTION

Sistema Assentensia

Purpose

The primary objective of the program is to effectively assess the applicant thoroughly through a completely automated system that not only helps you to save a lot of time but also gives quick outcomes. It is a cost-effective and well-known indicates of mass-assessment program.

Systema Asistencia will reduce the hectic job of taking attendances of the students.

Will reduce paper work, Can generate daily or weekly or monthly attendance reports instantly as and when required.

Systema Asistencia is designed for educational system institutes like schools, colleges and private institutes to take attendances of their students on regular basis.

This project will be very helpful for educational institutes where regular evaluation of student's is required. Further it can also be useful for anyone who requires feedback of regular present and absent.

Project Profile

Project Profile

Name	Sistma Assentensia
Institute	Atmiya University
Department	Department of Computer Science
Semester	Sem 3
Developed By	Dipen Pamar, Priyanka Maheta
Project Guide	Chiraj Jagani
Submitted to	Mscit

Atmiya University,

Sistma Assentensia - NodeJS | MSc IT, 2019

SPECIFICATION

Project Specification & Technical Details

Operating System Windows, Mac, Linux

Web Technology NodeJS

Web Server Apache Server

Front End HTML5, JS, Bootstrap

Back End Nodejs, TypeScript, PugJS

Database MySQL

Framework ExpressJs, Bootstrap

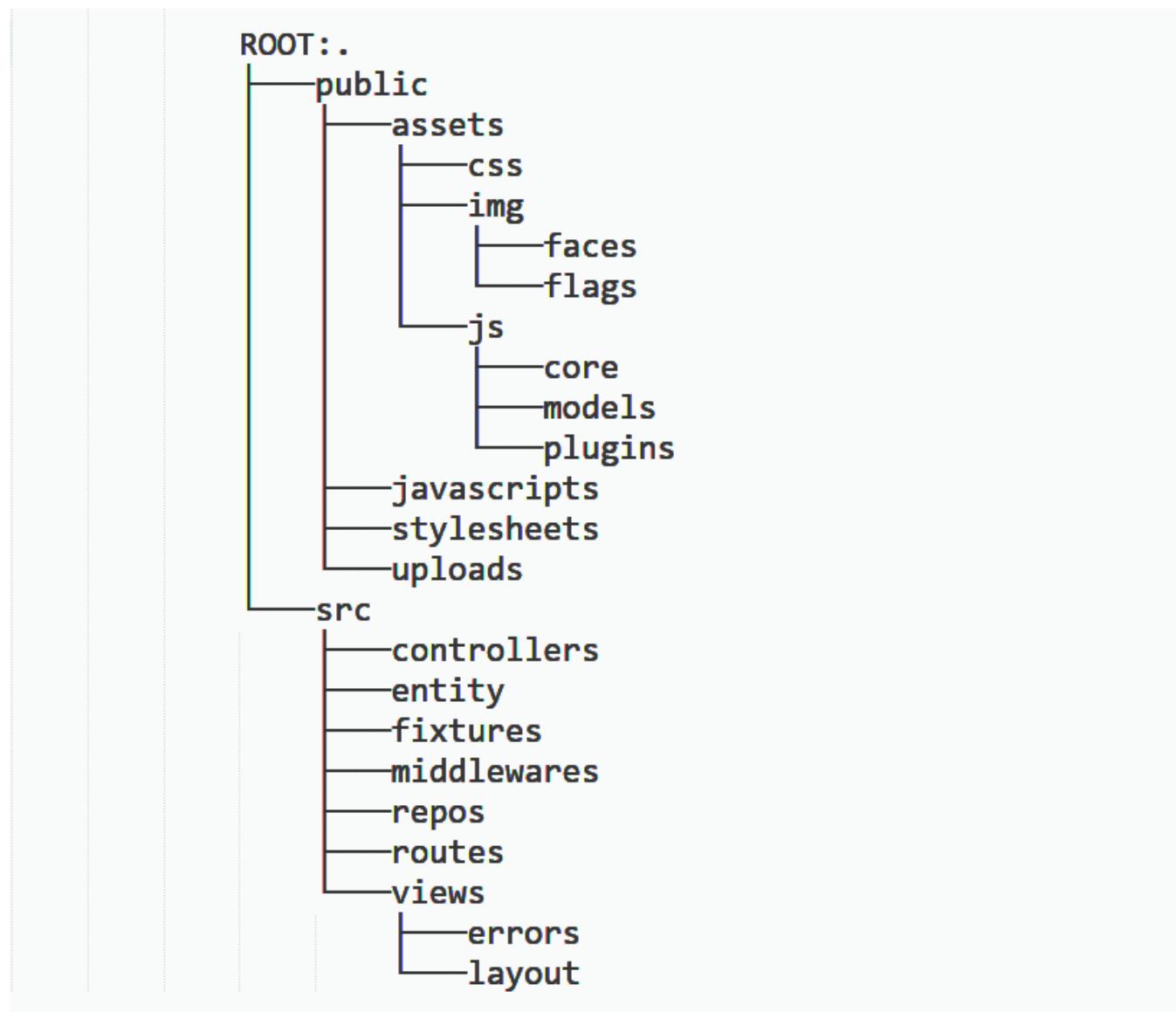
Libraries TypeORM, JWT, FaceAPI,

IDE, Text Editor Web Strome, VsCode

Dev Tools nodemon, ts-node-dev

Browser Chrome 50+, Firefox 40+

Project Directory Structure



SYSTEM ANALYSIS

Existing System

Existing system is a manual one in which users are maintaining books to store the information like Student Details, Instructor Details, Schedule Details and feedbacks about faculties who took attendances as per schedule. It is very difficult to maintain historical data.

Drawbacks of existing system:

- ✓ A lot of copies of attendance sheets have to be made.
- ✓ A lot of correction work hence delay in giving the reports of present and absent.
- ✓ A lot of tabulation work for daily attendances.
- ✓ Very time consuming.

Requirements of New System

Existing system is a paper based. It is very difficult to assets because instructor need to take attendance of all the students by preparing sheets. It is very difficult to main historic data. It is time consuming and costly as copies of paper have to made.

So to overcome all this disadvantages we required Eco-System which will reduce problems. And It not will be time consuming process.

Advantage Of New System

- ✓ It is ECO-FRIENDLY SYSTEM.
- ✓ Easy to maintain records.
- ✓ No need to assets paper.
- ✓ Allows automatic generation of attendance sheets in filtering forms.
- ✓ Generates report of present and absent of students.
- ✓ Allows you to take attendances by capturing and adding a photo..
- ✓ From anywhere user can access this site within internet, intranet.
- ✓ Totally responsive site.

User requirement:

- ✓ User should have technical knowledge to access this site.
- ✓ Individual Faculties must have individual device to take attendance

Feasibility Study:

The feasibility study is the important step in any software development process. This is because it makes analysis of different aspects like cost required for developing and executing the system, the time required for each phase of the system and so on. If these important factors are not analyzed then definitely it would have impact on the organization and the development and the system would be a total failure. So for running the application and the organization successfully this step is a very important step in a software development life cycle process.

(1) Technical Feasibility:-

If internet connection is slow than it will take time to load the site. User should have technical knowledge to access this site.

(2) Economic Feasibility:-

No extra charges are required to access this system. Due to hosting charges sometimes it becomes costly, it is totally depending upon which plan you are selecting and for how much time

(3) Operational Feasibility:-

It is very interactive and user friendly. Easy to navigate and access. This site is compatible will the devices like mobile, tablet, etc.)

(4) Social Feasibility:-

This type of feasibility studies are conducted to determine the degree to which the design or location of a given project is socially justified.

Use case

A use case describes how a user uses a system to accomplish a particular goal. A use case diagram consists of the system, the related use cases and actors and relates these to each other to visualize: what is being described? (System), who is using the system? (Actors) and what do the actors want to achieve? (Use cases), thus, use cases help ensure that the correct system is developed by capturing the requirements from the user's point of view.

What is a use case diagram?

In the Unified Modeling Language (UML), a use case diagram can summarize the details of your system's users (also known as actors) and their interactions with the system. To build one, you'll use a set of specialized symbols and connectors. An effective use case diagram can help your team discuss and represent:

- ✓ Scenarios in which your system or application interacts with people, organizations, or external systems
- ✓ Goals that your system or application helps those entities (known as actors) achieve
- ✓ The scope of your system

When to apply use case diagrams?

A use case diagram doesn't go into a lot of detail—for example, don't expect it to model the order in which steps are performed. Instead, a proper use case diagram depicts a high-level overview of the relationship between use cases, actors, and systems. Experts recommend that use case diagrams be used to supplement a more descriptive textual use case.

UML is the modeling toolkit that you can use to build your diagrams. Use cases are represented with a labeled oval shape. Stick figures represent actors in the process, and the actor's participation in the system is modeled with a line between the actor and use case. To depict the system boundary, draw a box around the use case itself.

UML use case diagrams are ideal for:

- ✓ Representing the goals of system-user interactions
- ✓ Defining and organizing functional requirements in a system
- ✓ Specifying the context and requirements of a system
- ✓ Modeling the basic flow of events in a use case

Use case diagram components

To answer the question, "What is a use case diagram?" you need to first understand its building blocks. Common components include:

Actors: The users that interact with a system. An actor can be a person, an organization, or an outside system that interacts with your application or system. They must be external objects that produce or consume data.

System: A specific sequence of actions and interactions between actors and the system. A system may also be referred to as a scenario.

Goals: The end result of most use cases. A successful diagram should describe the activities and variants used to reach the goal.

Use case diagram symbols and notation

The notation for a use case diagram is pretty straightforward and doesn't involve as many types of symbols as other UML diagrams.

- ✓ Use cases: Horizontally shaped ovals that represent the different uses that a user might have.
- ✓ Actors: Stick figures that represent the people actually employing the use cases.
- ✓ Associations: A line between actors and use cases. In complex diagrams, it is important to know which actors are associated with which use cases.

System:

Draw your system's boundaries using a rectangle that contains use cases. Place actors outside the system's boundaries.



Use Case:

Draw use cases using ovals. Label the ovals with verbs that represent the system's functions.



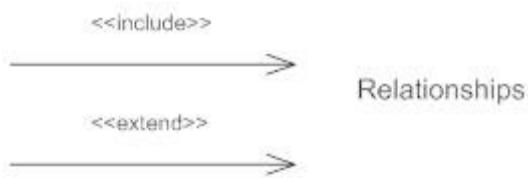
Actors:

Actors are the users of a system. When one system is the actor of another system, label the actor system with the actor stereotype.



Relationships:

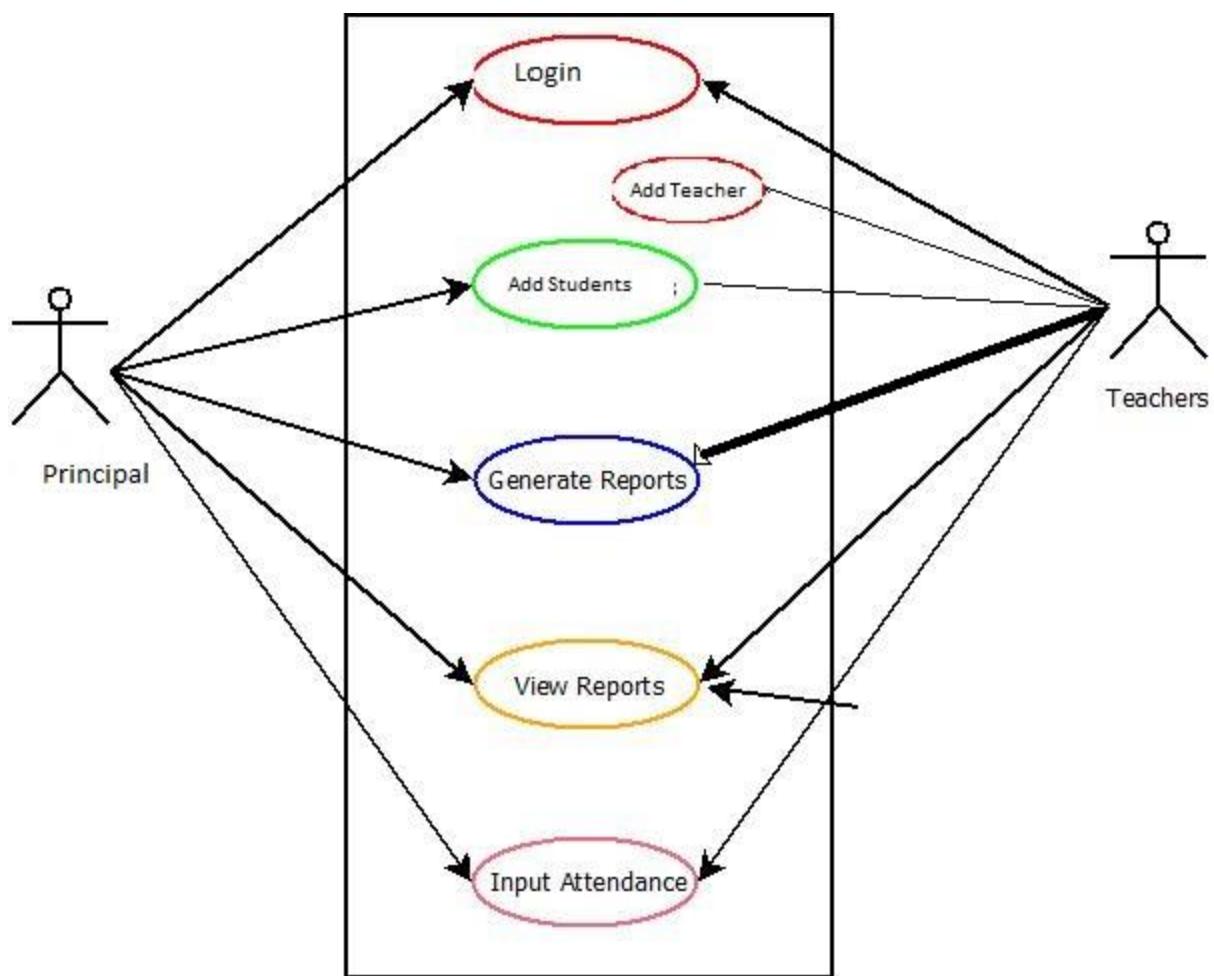
Illustrate relationships between an actor and a use case with a simple line. For relationships among use cases, use arrows labeled either "uses" or "extends." A "uses" relationship indicates that one use case is needed by another in order to perform a task. An "extends" relationship indicates alternative options under a certain use case.



Src 1: <https://www.lucidchart.com/pages/uml-use-case-diagram>

Src 2: <https://www.smartdraw.com/use-case-diagram>

Use case diagram for Attendance system



Drawing Tool Used- : **Draw.io**

Draw.io is free online diagram software for making flowcharts, process diagrams, org charts, UML, ER and network diagrams.

Src: <https://www.draw.io/>

NodeJS

What is Node.js?

- Node.js is an open source server environment
- Node.js is free
- Node.js runs on various platforms (Windows, Linux, Mac OS X, etc.)
- Node.js uses JavaScript on the server.

Why Node.js?

- Node.js uses asynchronous programming!
- Here is how PHP or ASP handles a file request:
 - Sends the task to the computer's file system.
 - Waits while the file system opens and reads the file.
 - Returns the content to the client.
 - Ready to handle the next request.
- Here is how Node.js handles a file request:
 - Sends the task to the computer's file system.
 - Ready to handle the next request.
 - When the file system has opened and read the file, the server returns the content to the client.

What Can Node.js Do?

- Node.js can generate dynamic page content
- Node.js can create, read, write, delete, and close files on the server
- Node.js can collect form data
- Node.js can add, delete, modify data in your database

What is a Node.js File?

- Node.js files contain tasks that will be executed on certain events
- A typical event is someone trying to access a port on the server
- Node.js files must be initiated on the server before having any effect
- Node.js files have extension ".js".

Node Package Manager

What is NPM?

- Node.js is an open source server environment
- Node.js is free
- Node.js runs on various platforms (Windows, Linux, Mac OS X, etc.)
- Node.js uses JavaScript on the server

More on NPM?

- NPM is a package manager for Node.js packages, or modules if you like.
- www.npmjs.com hosts thousands of free packages to download and use.
- The NPM program is installed on your computer when you install Node.js

What is a Package?

- A package in Node.js contains all the files you need for a module.
- Modules are JavaScript libraries you can include in your project.

Use NPM to

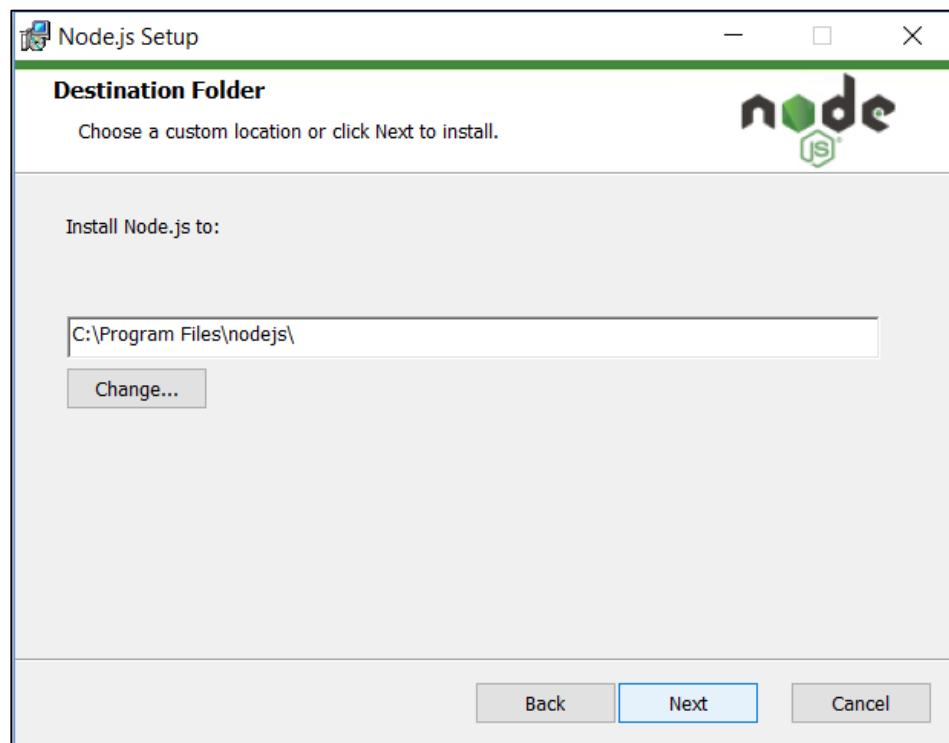
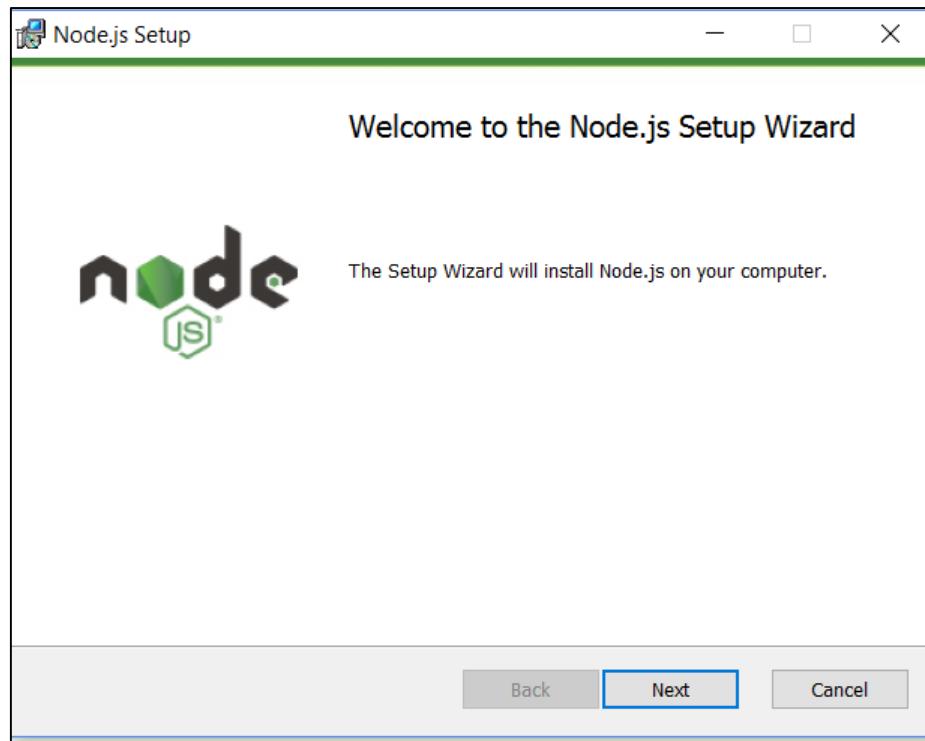
- Adapt packages of code for your apps, or incorporate packages as they are.
- Download standalone tools you can use right away.
- Run packages without downloading using npm.
- Share code with any npm user, anywhere.
- Restrict code to specific developers.
- Form virtual teams by using Orgs.
- Manage multiple versions of code and code dependencies.
- Update applications easily when underlying code is updated.
- Discover multiple ways to solve the same puzzle.
- Find other developers who are working on similar problems and projects.

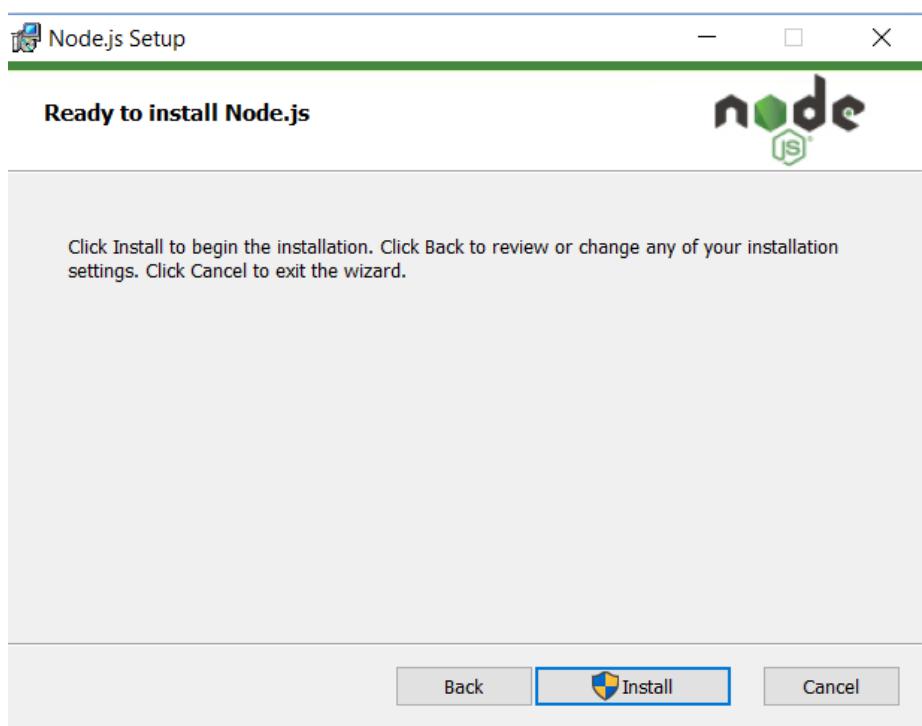
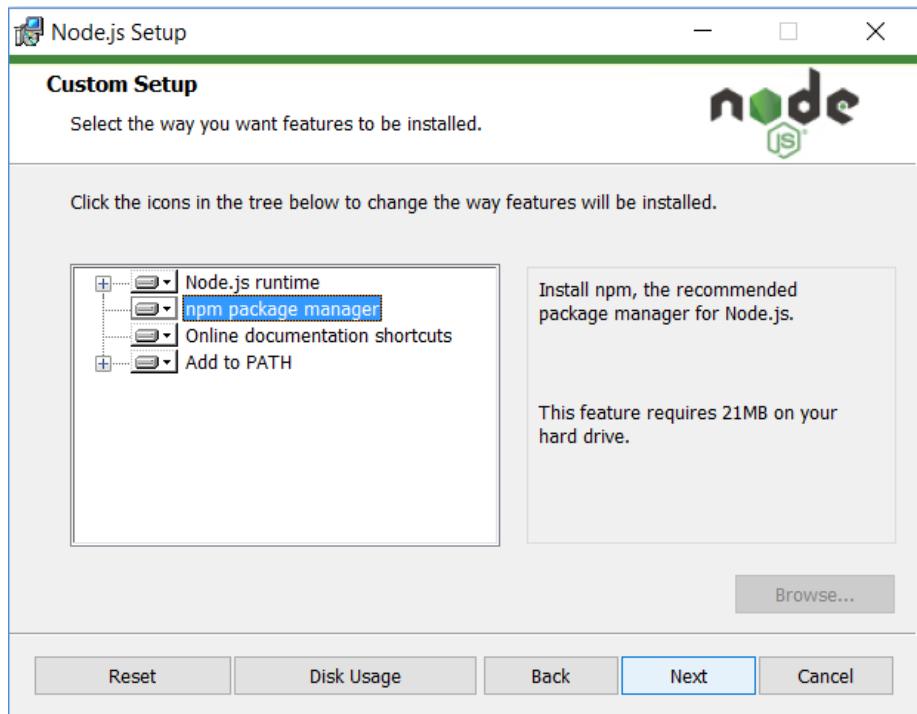
Install Node.js & NPM on Windows

Go to the **official Node.js site** which will offer you two versions of Node to download. Click on the Installer that says “Recommended for Most Users”.

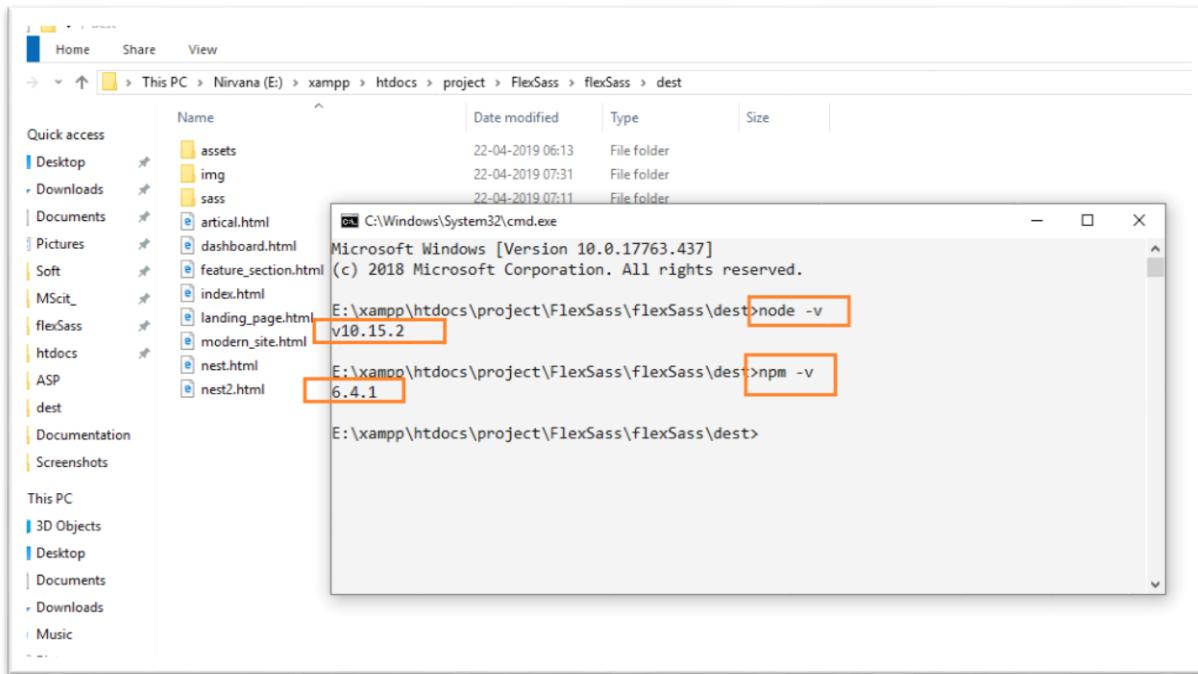
The screenshot shows the official Node.js website at <https://nodejs.org/en/>. The main content area features a dark header with the Node.js logo and navigation links for HOME, ABOUT, DOWNLOADS, DOCS, FOUNDATION, GET INVOLVED, SECURITY, and NEWS. Below the header, a text block explains that Node.js is a JavaScript runtime built on Chrome's V8 JavaScript engine, using an event-driven, non-blocking I/O model. It highlights the large package ecosystem, npm. A prominent section titled "Download for Windows (x64)" offers two main options: "v6.10.0 LTS" (labeled "Recommended For Most Users") and "v7.7.1 Current" (labeled "Latest Features"). Both buttons have a red border around them. Below these buttons are links for "Other Downloads", "Changelog", and "API Docs". A note below the buttons suggests looking at the LTS schedule. At the bottom of the page, there's a footer with links for "Report Node.js issue", "Report website issue", and "Get Help", along with copyright information for the Linux Foundation and Joyent.

This screenshot is identical to the one above, showing the official Node.js website at <https://nodejs.org/en/>. The "v6.10.0 LTS" button is highlighted with a red box. A red arrow points from the bottom of the previous screenshot to the "node-v6.10.0-x64.msi" link in the footer, specifically to the ".msi" file extension.





In The Command Prompt. To confirm Node is installed type **node -v** which will print the current version number. To confirm NPM is installed type **npm -v** which will print its current version number.



USER INTERFACE

What is User Interface Design?

User Interface (UI) Design is the link between users and your website. It includes the basic design elements that need to be present in order for someone to navigate your site and make decisions. It is the ever-evolving relationship between a person and the system that they are using. It includes the way that your website interacts with users, the overall design and how information is presented.

There are many different ways that you can look at user interface, but the basics always include the communication from a product to the user and vice versa1.

UI Design is all about structure, user manipulation and communication. This is one of the reasons it's so important that you pay close attention to it. It is the basic building blocks of how your website is set up and functions when visited by your target audience. If it doesn't go smoothly, problems tend to follow.

Some of the elements involved in User Interface Design include the input controls, navigational components, informational components and containers4. The devil is in the details, as they say. Buttons, lists, toggles, icons, breadcrumbs, tags and more also play a huge role. It is the way that you design your site from the back-end so that your audience can have a seamless, and enjoyable, front-end experience.

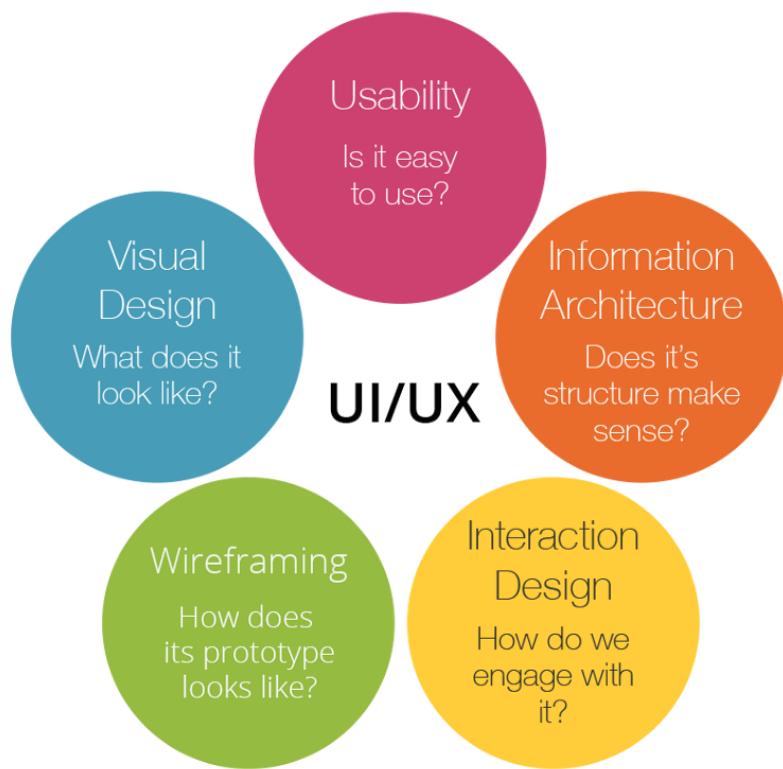
There are 3 issues / parts of a user interface design, these being

- how it looks,
- how it works,
- How easy it is to use.

Key Components of UI/UX Design

There are some core components of UX Design, which are as follows:

- Information Architecture
- Interaction Design
- Usability
- Wire framing
- Visual Design



Src: https://techterms.com/definition/user_interface

Interaction Design

Interaction Design is about creating the conceptual design with which the users interact with the product and/or application. This interaction includes various elements such as aesthetics, color, font, icons, images, motion, sound, space, graphics etc.

Usability

Usability can be referred as the user-friendliness as well. Along with figuring out if the users get the information they want by using the application or visiting the site for the first time and if the application/website is easy to navigate, the usability also comprehends the ways to handle the errors.

Wire framing

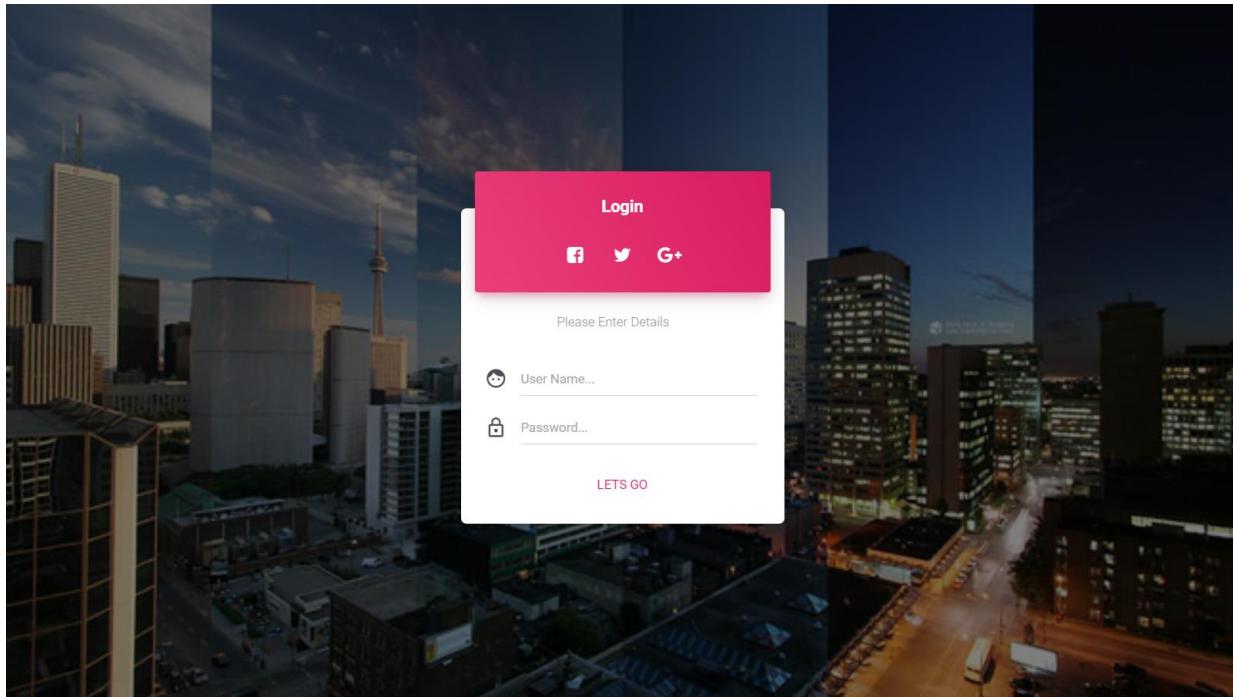
Wire framing is about creating a sample of the application to test the features, look and usability of the application before it is actually launched. It is a cheap way to test functionality and evaluate if the application serves the purpose.

Visual Design

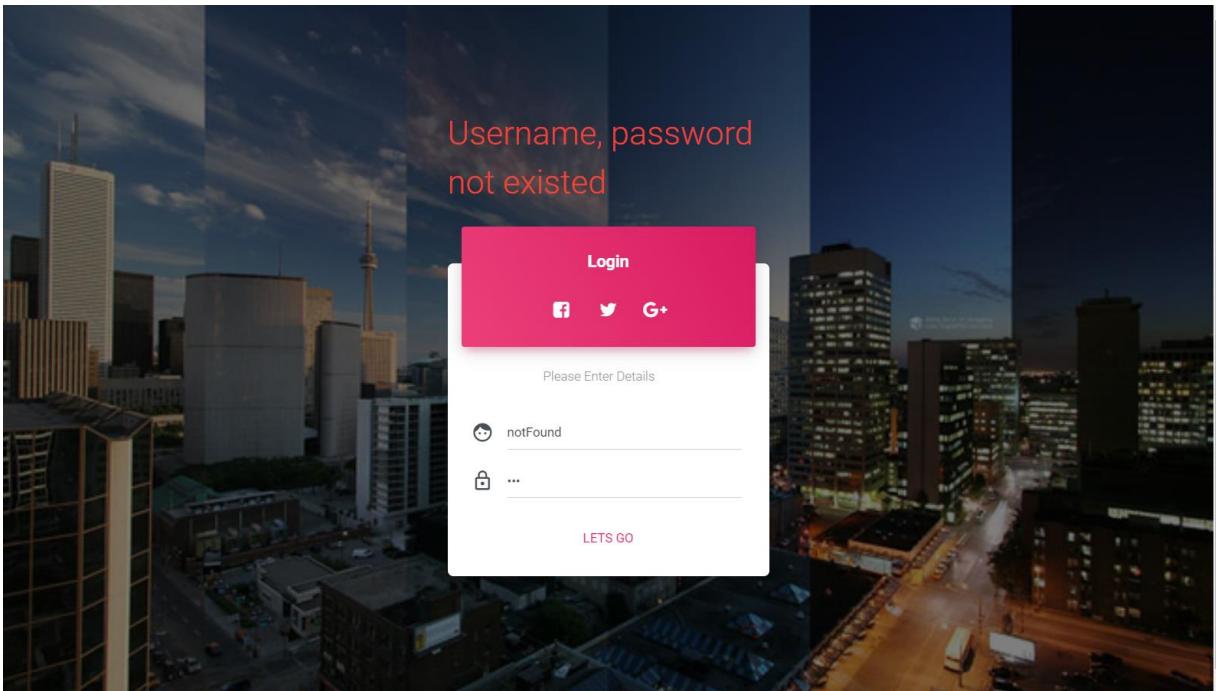
Visual design of the application or the website is like defining the company's brand. Finalizing the visual design can affect the users' behavior and hence it is the most important component of the UI design. Visual Design is not only about selecting the best images, colors, icons, fonts, but also identifying the appearance of the application impacts the interaction of its users. Now that you are aware with the UI/UX components, we would like to tell you how this whole procedure actually works.

Screenshots

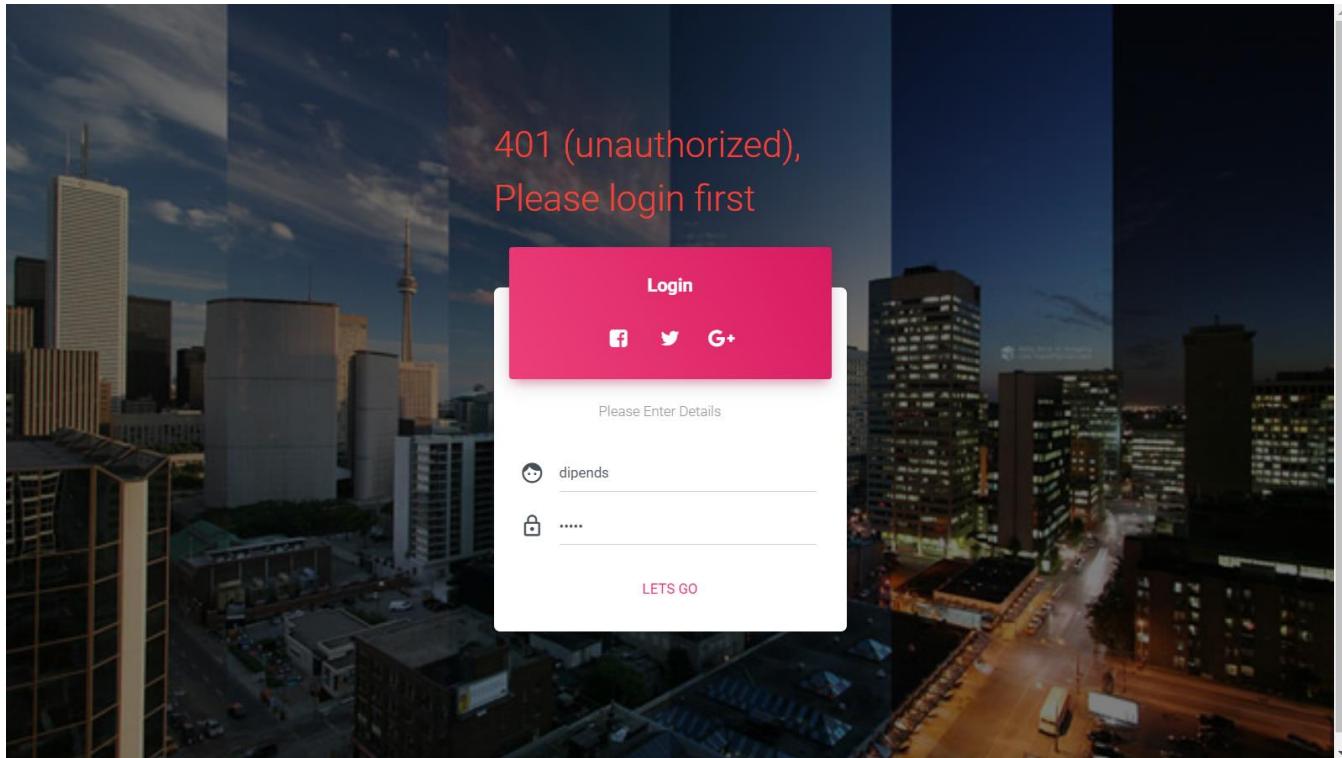
Login (Teacher, Principal)



User Must Enter Valid (Username & Password)



User Must be Logged in before go to dashboard



Teacher Registration Form (Principal Only)

AT

Teacher
Registration form

Username: TESTING

Password:

Full Name: Dipen Parmar

Email: dipensavji@gmail.com

Subject: Nodejs

Roll: principle

Submit

HOME ABOUT US BLOG LICENSES

©2019, made with ❤ Dipen Parmar for a better web.

Atmiya University,

Sistema Assentenia - NodeJS | MSc IT, 2019

Minimum Required fields (Validation)

The screenshot shows a registration form for a 'Teacher' account. The form includes fields for Username, Password, Full Name, Email, Subject, and Roll. The 'Username' and 'Password' fields are highlighted in red, indicating they are required. A red error message 'This field is required.' is displayed next to both fields. The other fields ('Full Name', 'Email', 'Subject', 'Roll') are shown in grey, suggesting they are optional or have been filled correctly.

Teacher
Registration form

Username for Login
This field is required.

Password Password must be strong
This field is required.

Full Name Your Good Name

Email your valid Email Address

Subject Name of Subject

Roll principle

CREATE

New Teacher Created Successfully notification

The screenshot shows a registration form for a 'Teacher' account. A green notification bar at the top left says 'Teacher Successfully created' with a close button. The rest of the form is identical to the one above, with fields for Username, Password, Full Name, Email, Subject, and Roll. The 'Password' field is currently empty, indicated by a purple placeholder line. The 'Submit' button is visible at the bottom of the form.

Teacher Successfully created

Teacher
Registration form

Username

Password

Full Name Your Good Name

Email your valid Email Address

Subject Name of Subject

Roll Admin | principle | Teacher

Submit

Student Registration form

The screenshot shows a web application interface. On the left is a sidebar with a dark background and white icons. It includes a user profile icon labeled 'mscit', a 'Attendance' icon, a 'Dashboard' icon, an 'AL_Attendance' icon, and a blue 'Add Student' button. Below these are 'Menu_sub' and 'Calendar' icons. At the bottom of the sidebar are links for 'HOME', 'ABOUT US', 'BLOG', and 'LICENSES'. The main content area has a purple header bar with the text 'Student' and 'Registration form'. Below the header is a form with fields for 'Full Name', 'Email address', 'Address' (with a placeholder 'Enter Your Full address seprated by comma (city, town, etc)'), 'mobile', 'enrollment_no', 'dd/mm/yyyy', 'RollNo', 'ProfoliPic' (with a 'Browse' button), and a 'Submit' button. A small note at the bottom right of the form says 'dd/mm/yyyy'. At the very bottom of the page, there is footer text: '©2019, made with ❤ Dipen Parmar for a better web.'

Validations for Student Registration Form

This screenshot is similar to the previous one but shows validation errors. A modal dialog box is displayed in the center, showing the message 'localhost:8000 says' followed by 'Upload Gif or Jpg images only' and an 'OK' button. The main form fields now have red error messages below them: 'Full Name' (This field is required.), 'mobile' (This field is required.), 'enrollment_no' (This field is required.), and 'RollNo' (This field is required.). The rest of the interface remains the same, including the sidebar and footer.

Submit Attendance, Select Absent/Present students by clicking row

The screenshot shows the 'Attendance' section of a web application. On the left is a sidebar with icons for Dashboard, ALAttendance, Add Student, Menu_sub, and Calendar. The main area has a header 'Student List' and 'Attendance Dashboard'. Below is a table with columns: Temp, Sr, img, Full Name, Roll, and Actions. The table rows are color-coded: green (rows 1, 3, 5), pink (rows 2, 4), and light green (row 6). Each row contains a small profile picture, a student ID, name, roll number, and a blue circular icon with a white symbol in the 'Actions' column. A green 'SUBMIT' button is at the bottom left. At the top right are search and user icons.

Temp	Sr	img	Full Name	Roll	Actions
●○	1		Priyanka	2	
○●	2		Nishit	4	
●○	3		PrakashSir	5	
○●	4		Riaz	3	
○○	5		Dipends	1	

Select Date & cross verify present Students & click on Conform button

This screenshot shows a modal dialog titled 'Present Students' over the main dashboard. The dialog asks 'Please select Date for Attendance?' and displays a date picker set to '04-10-2019'. Below the date picker is a table of student names: 1 dipends, 2 Jeel, 3 NIMIT, 4 PU, 5 NEO, and 6 RIAZ. At the bottom of the dialog are 'CANCEL' and 'CONFORM' buttons. The background shows the same attendance table as the previous screenshot, with the 4th row (Riaz) highlighted in blue to indicate it is selected.

Atmiya University,

Sistema Assentensia - NodeJS | MSc IT, 2019

Attendance successfully inserted Notification

The screenshot shows the Attendance Dashboard. At the top, there is a green notification bar with the text "Attendance Successfully Submitted.... Thank you for using our service". Below the notification, the title "Attendance Dashboard" is displayed. There are two buttons: "ADD STUDENT" and "ABSENT ALL". A table lists five students with their details: Sr, img, Full Name, Roll, and Actions. The students are: 1. Nishu (Roll 3), 2. PRAKASHSIR (Roll 5), 3. dipends (Roll 1), 4. Pri (Roll 7), and 5. Namu (Roll 6). Each row has an "Actions" column with a delete icon.

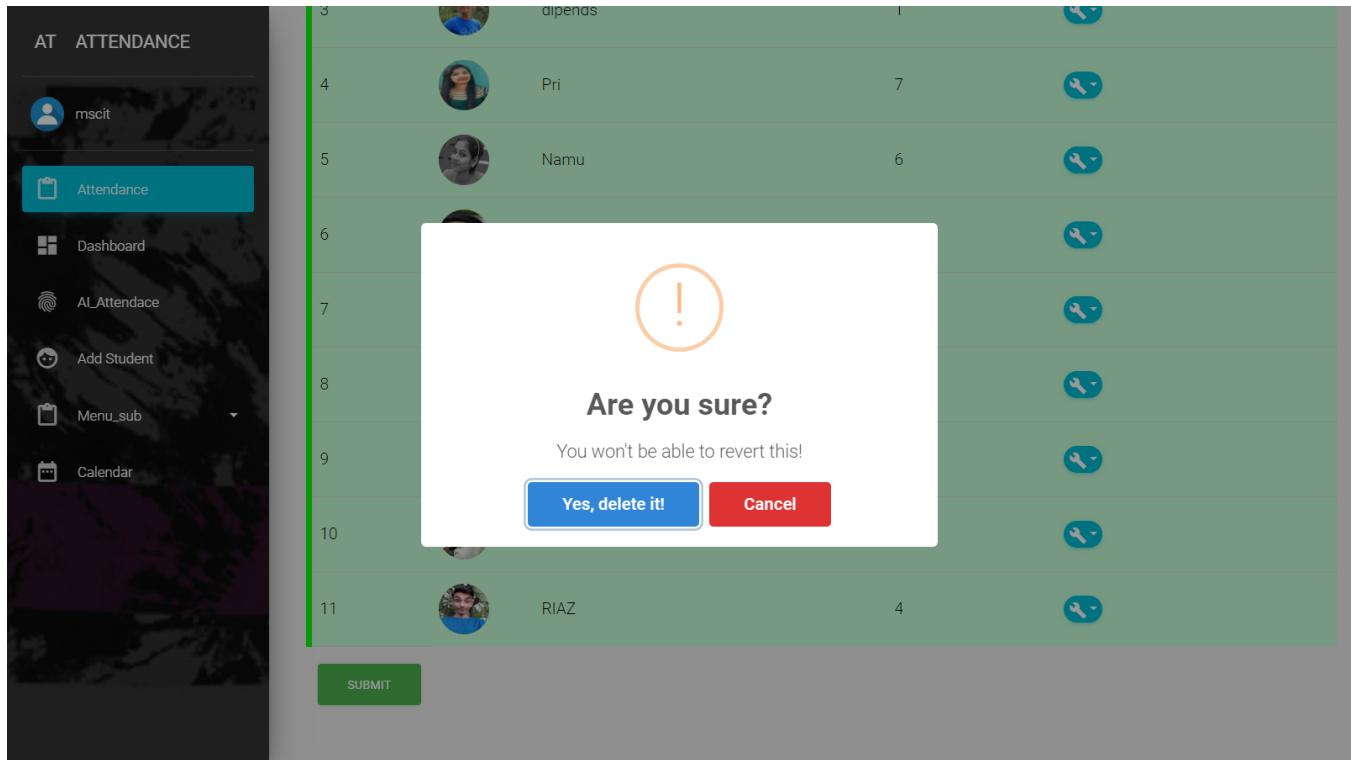
Sr	img	Full Name	Roll	Actions
1		Nishu	3	
2		PRAKASHSIR	5	
3		dipends	1	
4		Pri	7	
5		Namu	6	

Remove Students in Action column, by clicking Delete button

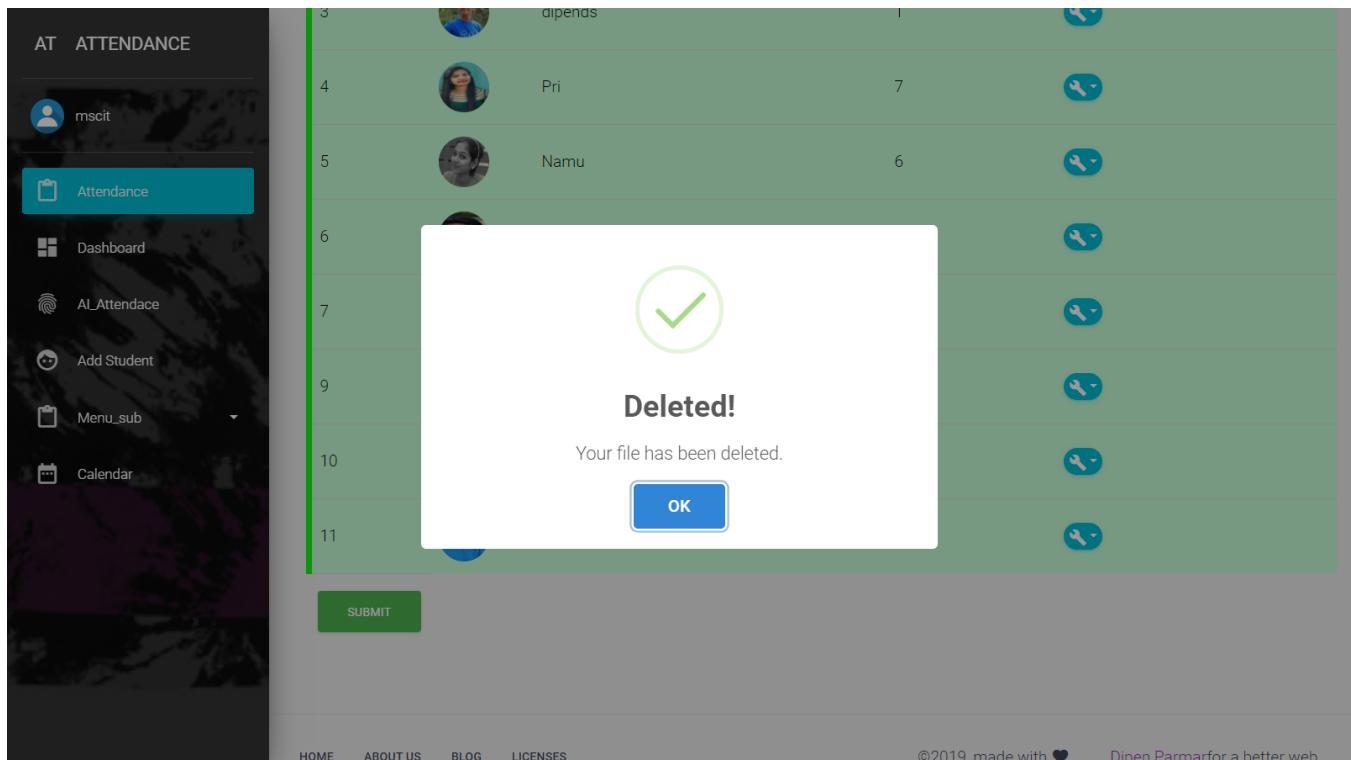
The screenshot shows the Attendance Dashboard. A student named "dipen_Testing" (Roll 12) is selected, indicated by a red background. A context menu is open over this row, showing "View" and "Delete" options. The "Delete" option is highlighted with a purple background. The rest of the student list is visible below, including students like PU (Roll 10), NEO (Roll 8), and RIAZ (Roll 4).

3		dipends	1	
4		Pri	7	
5		Namu	6	
6		Jeel	9	
7		NIMIT	11	
8		dipen_Testing	12	
9		PU	10	
10		NEO	8	
11		RIAZ	4	

Conformation notification for completing removing student from Database



Student Removed from Database Success notification



Dashboard Fixed Layout (Present, Absent Both)

The screenshot shows a fixed-layout dashboard titled "DashBoard". On the left is a sidebar with a dark background and white icons. The main area has a light gray header with the title "DashBoard" and two tabs: "STATIC" and "DYNAMIC". Below the tabs is a table showing student attendance data. The table has columns for date (2019-10-04, 2019-10-11), absent_present (Present, Absent, Present), and Totals. The data is as follows:

student.name	date	absent_present		Totals
		Present	Absent	
Jeel	2019-10-04	1	1	2
NEO	2019-10-04	1	1	2
NIMIT	2019-10-04	1	1	2
Namu	2019-10-04	1	1	2
Nishu	2019-10-04	1	1	2
PRAKASHSIR	2019-10-04	1	1	2
PU	2019-10-04	1	1	2
Pri	2019-10-04	1	1	2
RIAZ	2019-10-04	1	1	2
dipends	2019-10-04	1	1	2
	Totals	10	8	20

At the bottom of the dashboard, there are links for HOME, ABOUT US, RING, LIFESPAN, and a copyright notice: ©2019 made with Node.js. Please use a better web.

Dynamic Dashboard where user can change, filter values just like Pivot table in Excel

The screenshot shows a dynamic dashboard titled "DashBoard". The sidebar and main layout are similar to the first dashboard. The main area now features a table with a "Table" header and a "Count" input field. To the right of the table is a filter section with dropdowns for "date" and "absent_present". The table data is identical to the first dashboard. The filtering sidebar on the right lists various student attributes: id, student_id, teacher_id, present, createdAt, updatedAt, student.id, student.roll_no, student.enrollment_no, student.email, student.mobile, student.address, and student.profile_pic.

User Friendly UI

The screenshot shows a user-friendly web application interface. On the left is a dark sidebar menu with the following items:

- AT ATTENDANCE
- mscit
- Attendance
- Dashboard** (highlighted in blue)
- AL_Attendance
- Add Student
- Menu_sub
- Calendar

The main content area displays a table of attendance data:

	student.name	Jeel	NEO	NIMIT	Namu	Nishu	PRAKASHSIR	PU	Pri	dipends	null	Jeel	NEO	NIMIT	Namu	Nishu	PRAKASHSIR	PU	Pri	RIAZ	dipends	Totals		
date																								
2019-10-04		2		2	1	3		3		1	2	2	2	4	2	3	1	1	1	4	3	2	42	
2019-10-11		1	1	1	1	1		1		1	1									1	1		10	
Totals		3	1	3	2	4		4	1	2	2	2	2	4	2	3	1	1	1	4	3	5	3	52

At the bottom of the page, there is a footer note: ©2019, made with ❤ Dipen Parmar for a better web.

Take Attendance Group photo (AI Attendance) Select Choose file, upload Image

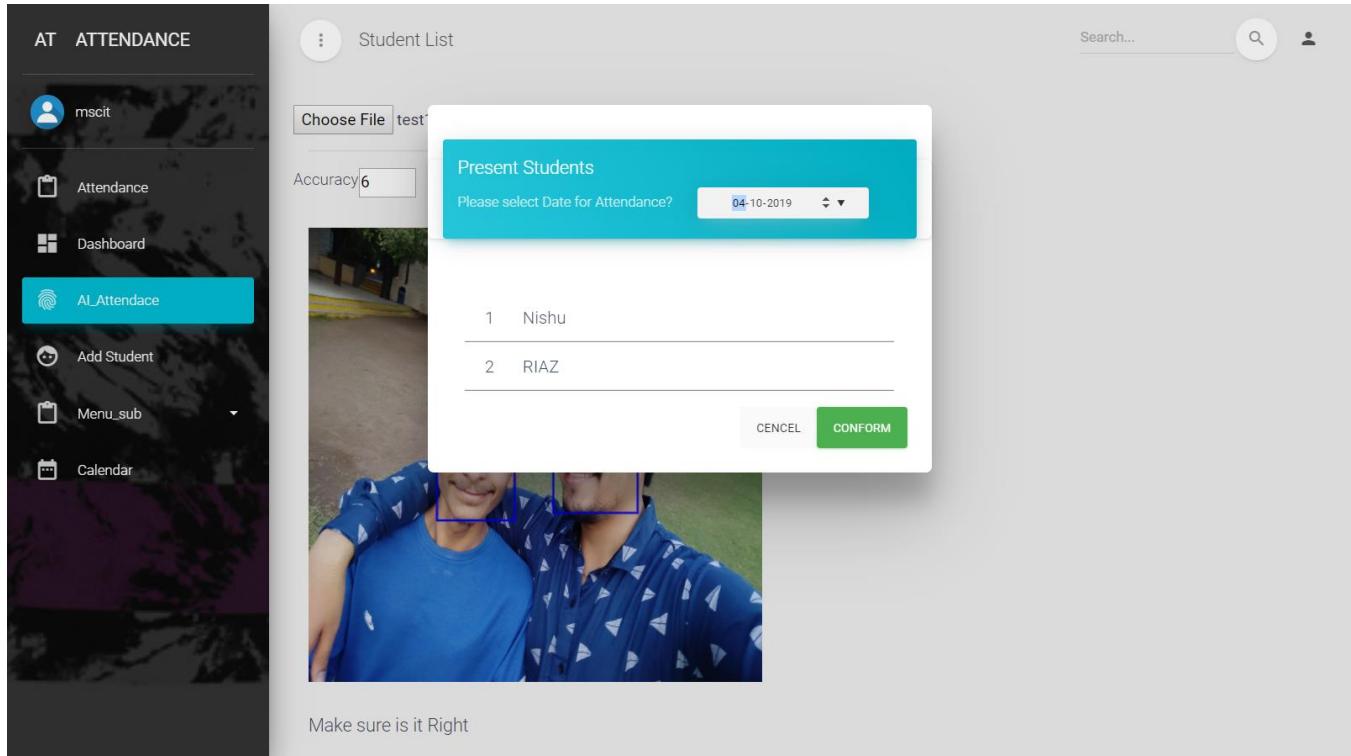
The screenshot shows the AI Attendance feature. The sidebar menu is identical to the previous screenshot. The main content area has the following elements:

- A "Student List" button with a three-dot menu icon.
- A search bar with a magnifying glass icon and a placeholder "Search...".
- A "Choose File" button with the text "No file chosen".
- An "Accuracy" input field containing the number "6".
- A placeholder message at the bottom: "Make sure is it Right"

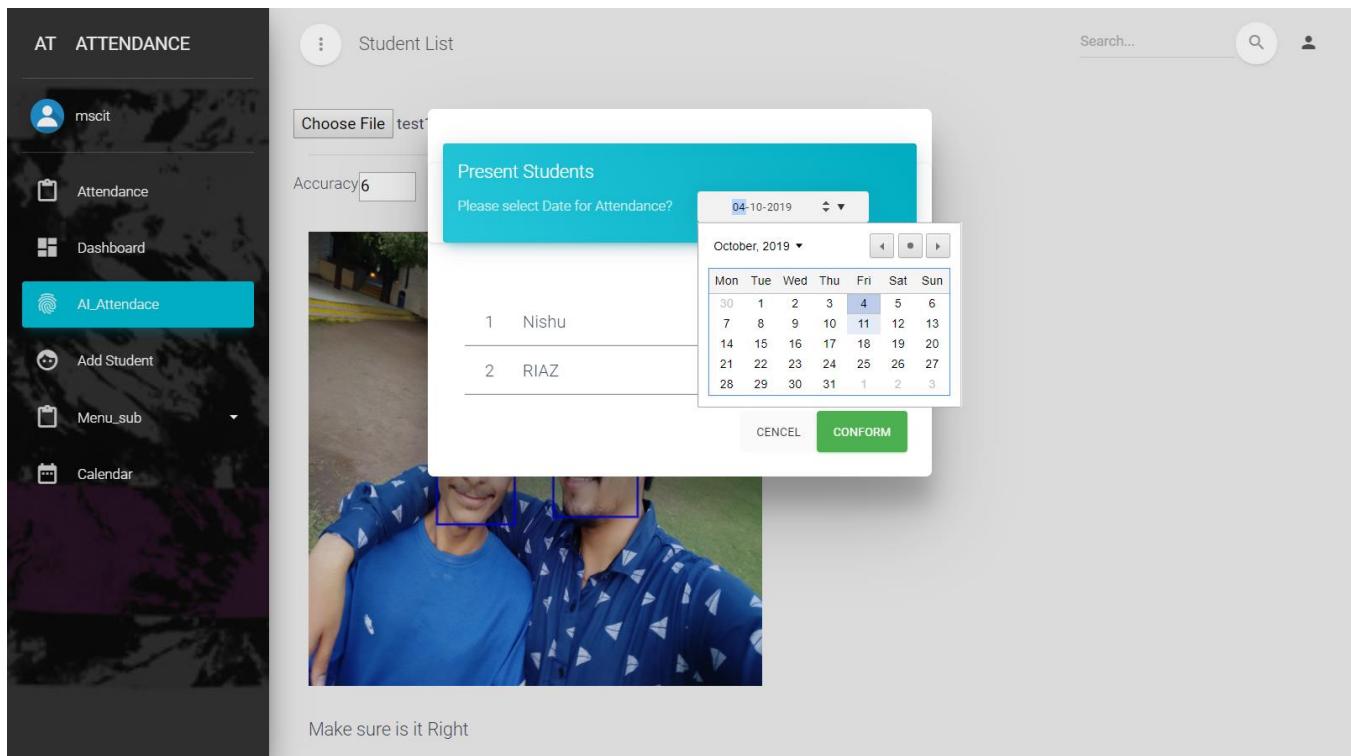
Atmiya University,

Sistema Assentensia - NodeJS | MSc IT, 2019

System will Recognition all students existed in uploaded Image by FaceAPI



Select Date & click on conform button



Atmiya University,

Sistema Assentencia - NodeJS | MSc IT, 2019

Log out (Signout)

The screenshot shows the Attendance Dashboard. On the left, there is a sidebar with a dark background and a purple header. The sidebar contains icons for Attendance, Dashboard, AI Attendance, Add Student, Menu_sub, and Calendar. At the bottom of the sidebar, it says "localhost:8000/logout". The main area has a light gray background. At the top, there is a navigation bar with "Student List" and a search bar. A profile dropdown menu is open, showing "Profile", "Settings", and a purple "Log out" button. Below the navigation is a section titled "Attendance Dashboard" with two buttons: "ADD STUDENT" and "ABSENT ALL". A table lists student information:

Sr	img	Full Name	Roll	Actions
1		Nishu	3	
2		PRAKASHSIR	5	
3		dipends	1	
4		Pri	7	
5		Namu	6	
6		Jeel	9	
7		KIRANIT	11	

The screenshot shows the Attendance Dashboard. On the left, there is a sidebar with a dark background and a grey header. The sidebar contains icons for Attendance, Dashboard, AI Attendance, Add Student, Menu_sub, and Calendar. At the bottom of the sidebar, it says "localhost:8000/logout". The main area has a light gray background. At the top, there is a navigation bar with "Student List" and a search bar. A profile dropdown menu is open, showing "My Profile" and a "Logout" button. Below the navigation is a section titled "Attendance Dashboard" with two buttons: "ADD STUDENT" and "ABSENT ALL". A table lists student information:

Sr	img	Full Name	Roll	Actions
1		Nishu	3	
2		PRAKASHSIR	5	
3		dipends	1	
4		Pri	7	
5		Namu	6	
6		Jeel	9	
7		KIRANIT	11	

Atmiya University,

Sistema Assentensia - NodeJS | MSc IT, 2019

Data Flow Diagram – DFD

A data flow diagram shows the way information flows through a process or system. It includes data inputs and outputs, data stores, and the various sub processes the data moves through. DFDs are built using standardized symbols and notation to describe various entities and their relationships.

Data flow diagrams visually represent systems and processes that would be hard to describe in a chunk of text. You can use these diagrams to map out an existing system and make it better or to plan out a new system for implementation. Visualizing each element makes it easy to identify inefficiencies and produce the best possible system.

Physical and logical data flow diagrams

Before actually creating your data flow diagram, you'll need to determine whether a physical or logical DFD best suits your needs. If you're new to data flow diagrams, don't worry—the distinction is pretty straightforward.

Logical data flow diagrams focus on what happens in a particular information flow: what information is being transmitted, what entities are receiving that info, what general processes occur, etc. The processes described in a logical DFD are business activities—a logical DFD doesn't delve into the technical aspects of a process or system. Non-technical employees should be able to understand these diagrams.

Physical data flow diagrams focus on how things happen in an information flow. These diagrams specify the software, hardware, files, and people involved in an information flow. A detailed physical data flow diagram can facilitate the development of the code needed to implement a data system.

Symbols and Notations Used in DFDs

Two common systems of symbols are named after their creators:

- ✓ Yourdon and Coad
- ✓ Yourdon and DeMarco
- ✓ Gane and Sarson

One main difference in their symbols is that Yourdon-Coad and Yourdon-DeMarco use circles for processes, while Gane and Sarson use rectangles with rounded corners, sometimes called lozenges. There are other symbol variations

in use as well, so the important thing to keep in mind is to be clear and consistent in the shapes and notations you use to communicate and collaborate with others.

Using any convention's DFD rules or guidelines, the symbols depict the four components of data flow diagrams.

1. **External entity:** an outside system that sends or receives data, communicating with the system being diagrammed. They are the sources and destinations of information entering or leaving the system. They might be an outside organization or person, a computer system or a business system. They are also known as terminators, sources and sinks or actors. They are typically drawn on the edges of the diagram.
2. **Process:** any process that changes the data, producing an output. It might perform computations, or sort data based on logic, or direct the data flow based on business rules. A short label is used to describe the process, such as "Submit payment."
3. **Data store:** files or repositories that hold information for later use, such as a database table or a membership form. Each data store receives a simple label, such as "Orders."
4. **Data flow:** the route that data takes between the external entities, processes and data stores. It portrays the interface between the other components and is shown with arrows, typically labeled with a short data name, like "Billing details."

Notation	Yourdon and Coad	Gane and Sarson
External Entity		
Process		
Data Store		
Data Flow		

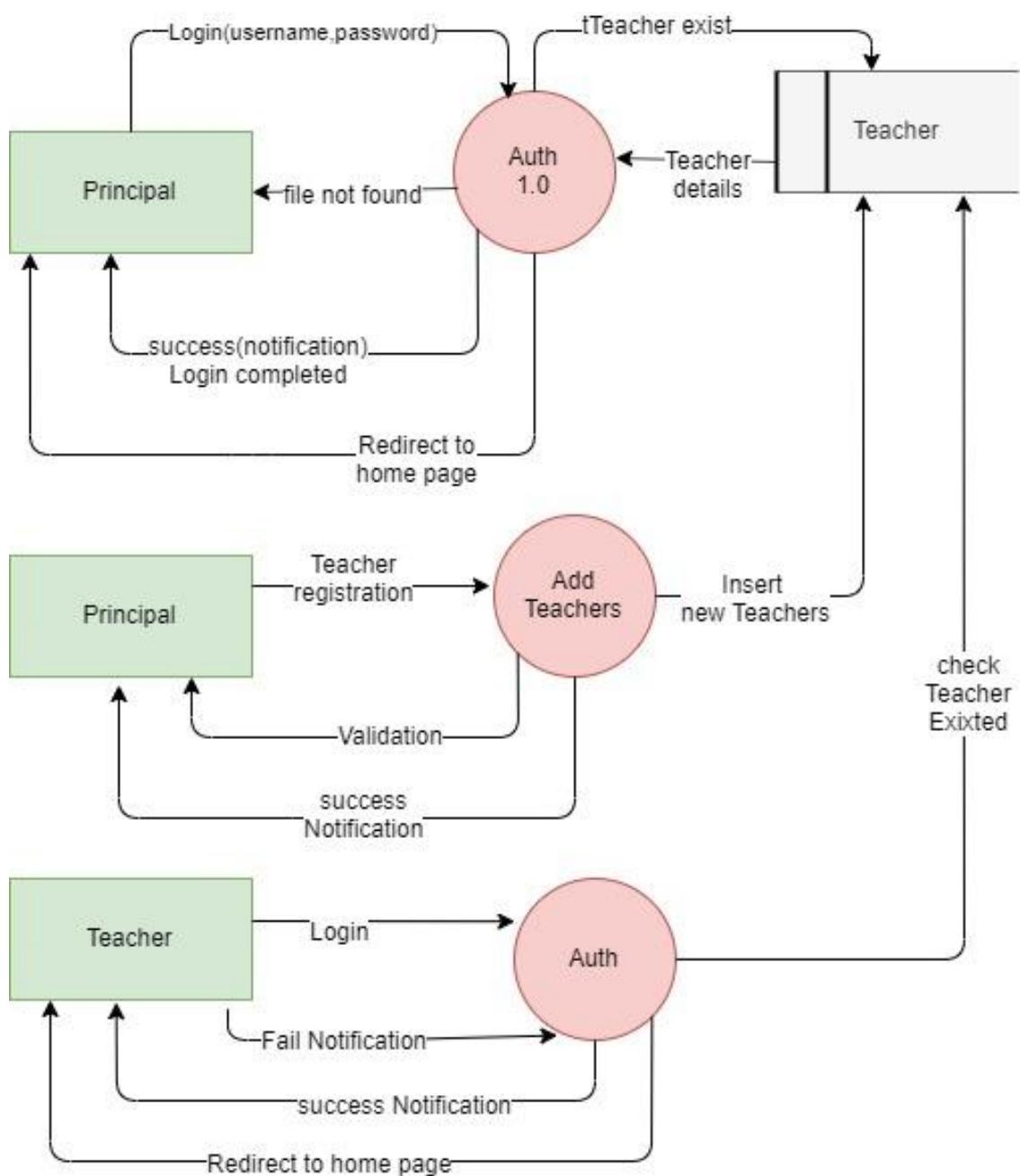
DFD rules and tips

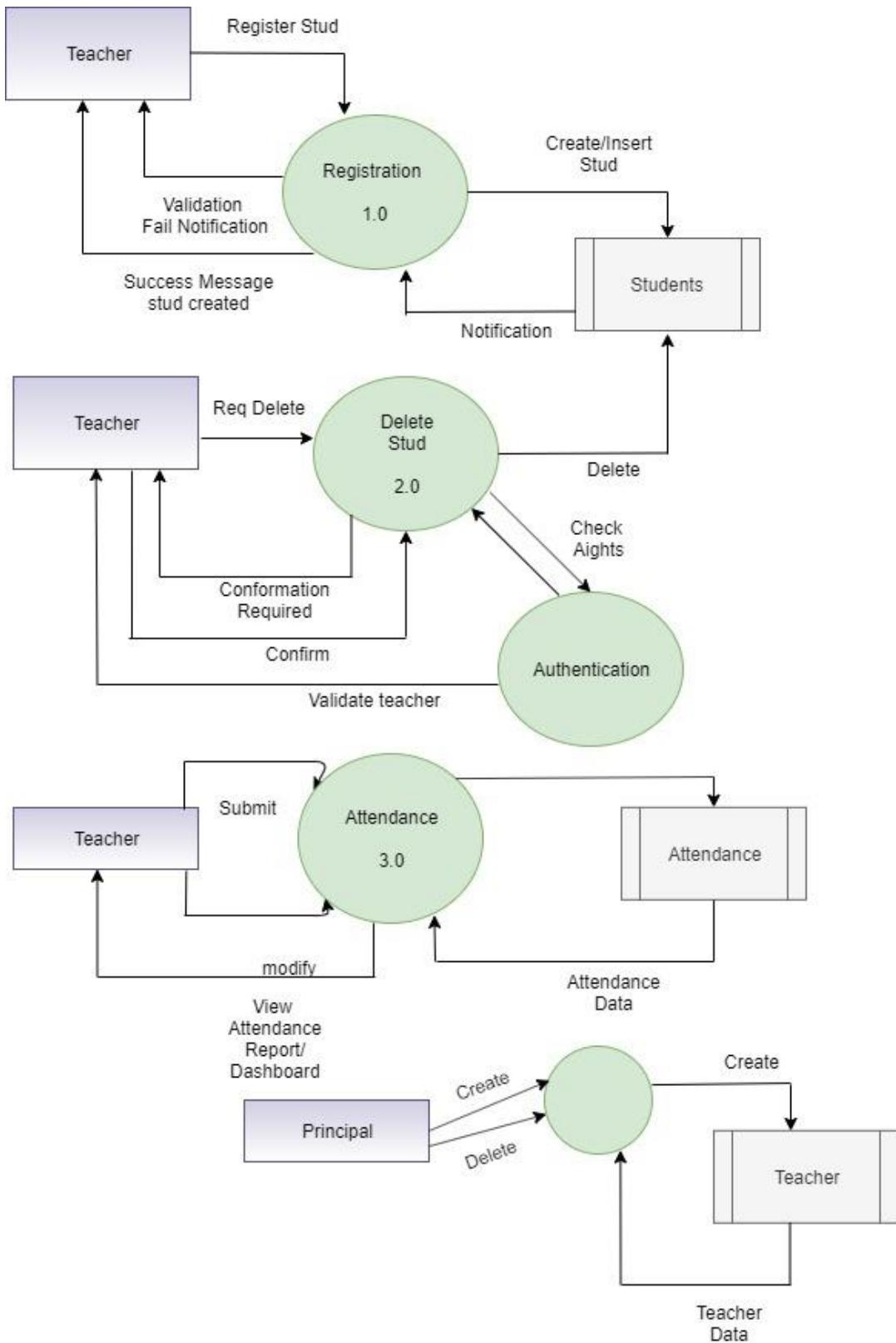
- ✓ Each process should have at least one input and an output.
- ✓ Each data store should have at least one data flow in and one data flow out.
- ✓ Data stored in a system must go through a process.
- ✓ All processes in a DFD go to another process or a data store.

Src: <https://www.lucidchart.com/pages/data-flow-diagram>

Src: <https://www.lucidchart.com/blog/data-flow-diagram-tutorial>

Data flow diagram Systema Asistencia





Data Dictionary

Data dictionary: is a file or a set of files that contains a database's metadata. The data dictionary contains records about other objects in the database, such as data ownership, data relationships to other objects, and other data.

The data dictionary is a crucial component of any relational database. Ironically, because of its importance, it is invisible to most database users. Typically, only database administrators interact with the data dictionary.

In a relational database, the metadata in the data dictionary includes the following:

- ✓ Names of all tables in the database and their owners
- ✓ Names of all indexes and the columns to which the tables in those indexes relate
- ✓ Constraints defined on tables, including primary keys, foreign-key relationships to other tables, and not-null constraints

For most relational database management systems (RDBMS), the database management system software needs the data dictionary to access the data within a database. For example, the Oracle DB software has to read and write to an Oracle DB. However, it can only do this via the data dictionary created for that particular database.

Src: <https://www.techopedia.com/definition/27752/data-dictionary>

Teachers:

SELECT * FROM `teacher`													Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]	
	Show all	Number of rows:	25	Filter rows:	Search this table	Sort by key:	None	Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]						
+ Options		id	username	password	name	subject	email	mobile	address	profile_pic	roll	dep_id	is_logged	createdAt
Edit	Copy	Delete	1629b32-4f02-4900-a190-3a4404e4ddc	dipen1	\$2a\$12\$97Qf/5UJfYpmIewYbxUw9OwH4oysY.Tdg F8fkTwOaC...	Teacher	nodejs	NULL	NULL	default.jpg	teacher	NULL	NULL	2015-09-04 00:00:00
Edit	Copy	Delete	4900-4b40-8627-d610c9424608	admin	\$2a\$12\$97Qf/5UJfYpmIewYbxUw9OwH4oysY.Tdg F8fkTwOaC...	admin	admin@admin	NULL	NULL	default.jpg	admin	NULL	NULL	2008-02-08 00:00:00
Edit	Copy	Delete	4ca1-48d1-93da-5653e35b5b7	user1	\$2a\$12\$97Qf/5UJfYpmIewYbxUw9OwH4oysY.Tdg F8fkTwOaC...	Teacher	nodejs	NULL	NULL	default.jpg	principle	NULL	NULL	0000-00-00 00:00:00
Edit	Copy	Delete	77682df9-ab38d1a1095	teacher	\$2a\$12\$97Qf/5UJfYpmIewYbxUw9OwH4oysY.Tdg F8fkTwOaC...	Teacher	nodejs	NULL	NULL	default.jpg	principle	NULL	NULL	0000-00-00 00:00:00
Edit	Copy	Delete	5c5eaf148a7a	dipen	\$2a\$12\$97Qf/5UJfYpmIewYbxUw9OwH4oysY.Tdg F8fkTwOaC...	Teacher	nodejs	NULL	NULL	default.jpg	principle	NULL	NULL	2015-09-09 00:00:00
Edit	Copy	Delete	8d983d0e-2224-4801-833e-513cf5fa4c31	p	\$2a\$12\$60DDR40vlk3djGzUAal8a.CtywqbhnR72A021cUh15...			NULL	NULL	default.jpg	principle	NULL	NULL	2019-08-31 14:59:59
Edit	Copy	Delete	dipen2	\$2a\$12\$97Qf/5UJfYpmIewYbxUw9OwH4oysY.Tdg F8fkTwOaC...	Teacher	nodejs		NULL	NULL	default.jpg	teacher	NULL	NULL	0000-00-00 00:00:00

Students:

SELECT * FROM `student`													Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]	
	Show all	Number of rows:	25	Filter rows:	Search this table	Sort by key:	None	Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]						
+ Options		id	roll_no	enrollment_no	name	subject	email	mobile	address	profile_pic	createdAt	updatedAt	teacherId	
Edit	Copy	Delete	020785ed-8c15-47e8-83af-a9e27977433	12	22	namu	NULL		default.jpg	2019-08-31 15:08:52.902501	2019-08-31 15:08:52.902501	a190-3a4404e4ddc		
Edit	Copy	Delete	439fb9a-84d8-4821-a520-814635829e2	9	9	piyudo	NULL		default.jpg	2019-08-31 15:03:45.399024	2019-08-31 15:03:45.399024	a190-3a4404e4ddc		
Edit	Copy	Delete	49983d02-3b34-40a7-b5f9-cb3ca5cc4304	4	4	a	NULL		default.jpg	2019-08-31 15:00:48.738507	2019-08-31 15:00:48.738507	1629b32-4f02-4900-a190-3a4404e4ddc		
Edit	Copy	Delete	73b243f4-e1b7-4c55-9567-349d06ebf3b	2	2	b	NULL		default.jpg	2019-08-31 14:56:55.350337	2019-08-31 14:56:55.350337	a190-3a4404e4ddc		
Edit	Copy	Delete	78a45888-ef3b-45ad-8b49-fa775c9a3c58	3	3	c	NULL		default.jpg	2019-08-31 14:57:05.135234	2019-08-31 14:57:05.135234	8d983d0e-2224-4801-833e-513cf5fa4c31		

Attendance:

SELECT * FROM attendance									Profiling Edit inline [Edit] [Explain SQL] [Create PHP code] [Refresh]	
		Show all	Number of rows:	25	Filter rows:	Search this table	Sort by key:	None		
+ Options	+ T	id	student_id	teacher_id	present	date	createdAt	updatedAt	studentId	
Edit	Copy	Delete	2f0995e-7418-4bc5-b558-38473d90e31	020785ed-8c15-47e8-83af-a6c927977433	16290b32-4f02-4900-a190-3a440f4c4cd	1	2019-08-15 05:30:00	2019-08-31 16:08:06.477150	2019-08-31 15:08:06.477150	020785ed-8c15-47e8-83af-a6c927977433
Edit	Copy	Delete	34004278-2004-4fc0-b251-4562cf7d9ab	cd005f6-f139-4bce-9ccf-526564ca52d6	8dd80de9-9db9-4548-9db3-600c7fbeda7d	1	2019-08-31 05:30:00	2019-08-31 14:57:55.268844	2019-08-31 14:57:55.268844	cd005f6-f139-4bce-9ccf-526564ca52d6
Edit	Copy	Delete	3a47f5052-0a20-40fa-8d7e-fc2b08c7eed8	ce92cb55-2a08-4b69-a656-1e053b510e92	bba7fe2-d420-4416-b2a6-4ad765ea083a	0	2019-08-26 05:30:00	2019-08-03 16:20:24.727067	2019-09-03 16:20:24.727067	ce92cb55-2a08-4b69-a656-1e053b510e92
Edit	Copy	Delete	6779397e-a83a-41bd-940e-a8781e20cfe	439ff9a-84a8-4821-a520-614a35382e2	16290b32-4f02-4900-a190-3a440f4c4cd	0	2019-08-24 05:30:00	2019-08-31 15:11:50.047425	2019-08-31 15:11:50.047425	439ff9a-84a8-4821-a520-614a35382e2
Edit	Copy	Delete	6b800939-d8ed-4019-a189-8a5d775ee2f	49903d02-3b34-440c-7b0f9-cb8da5c4304	16290b32-4f02-4900-a190-3a440f4c4cd	1	2019-08-24 05:30:00	2019-08-31 15:11:50.092305	2019-08-31 15:11:50.092305	49903d02-3b34-440c-7b0f9-cb8da5c4304
Edit	Copy	Delete	82fcd85b-b970-4675-8517-ab880208d8d	40903d02-3b34-40c7-b0f9-cb8da5c4304	16290b32-4f02-4900-a190-3a440f4c4cd	0	2019-08-15 05:30:00	2019-08-31 15:04:58.826977	2019-08-31 15:04:58.826977	40903d02-3b34-40c7-b0f9-cb8da5c4304
Edit	Copy	Delete	84462208-cb10-4a59-984e-8e923d31c1	439ff9a-84a8-4821-a520-61a435322e9	16290b32-4f02-4900-a190-3a440f4c4cd	0	2019-08-14 05:30:00	2019-08-31 15:03:52.817907	2019-08-31 15:03:52.817907	439ff9a-84a8-4821-a520-61a435322e9
Edit	Copy	Delete	8803efc-c934-427e-a00a-a2edf552daae	732c4394-e1b7-4c85-95e7-8459a88ebfb3	8dd80de9-9db9-4548-9db3-600c7fbeda7d	1	2019-08-31 05:30:00	2019-08-31 14:57:54.903377	2019-08-31 14:57:54.903377	732c4394-e1b7-4c85-95e7-8459a88ebfb3
Edit	Copy	Delete	98cc377c-c055-4b32-a4eb-aceb0b92ea0	49903d02-3b34-440c-7b0f9-cb8da5c4304	16290b32-4f02-4900-a190-3a440f4c4cd	1	2019-08-14 05:30:00	2019-08-31 15:03:52.828878	2019-08-31 15:03:52.828878	40903d02-3b34-40c7-b0f9-cb8da5c4304
Edit	Copy	Delete	a590cd94c-8e74-4470-bf06-fa7ac795d04	78e45868-dd3b-45ad-8b49-fa775c9a3c58	8dd80de9-9db9-4548-9db3-600c7fbeda7d	0	2019-08-31 05:30:00	2019-08-31 14:57:55.268844	2019-08-31 14:57:55.268844	78e45868-dd3b-45ad-8b49-fa775c9a3c58

PUG

What is Pug | Jade

Pug.js is a HTML templating engine, which means you can write much simpler Pug code, which Pug compiler will compile into HTML code, that browser can understand. ... Pug has powerful features like conditions, loops, includes, mixins using which we can render HTML code based on user input or reference data

Pug.js to make your life easier with HTML templates

Pug.js is a HTML templating engine, which means you can write much simpler Pug code, which Pug compiler will compile into HTML code, that browser can understand. But why not write HTML code in the first place?

Pug has powerful features like conditions, loops, includes, mixins using which we can render HTML code based on user input or reference data. Pug also support JavaScript natively, hence using JavaScript expressions, we can format HTML code.

Pug.js, as from its name is a JavaScript library. Hence, we can run it inside browser or on node.js. Here, I am using node.js to demonstrate few examples.

This Gulp plugin enables you to compile your Pug templates into HTML or JS, with support for template locals, custom Pug filters, AMD wrapping, and others. Here is a simple example using gulp-pug:

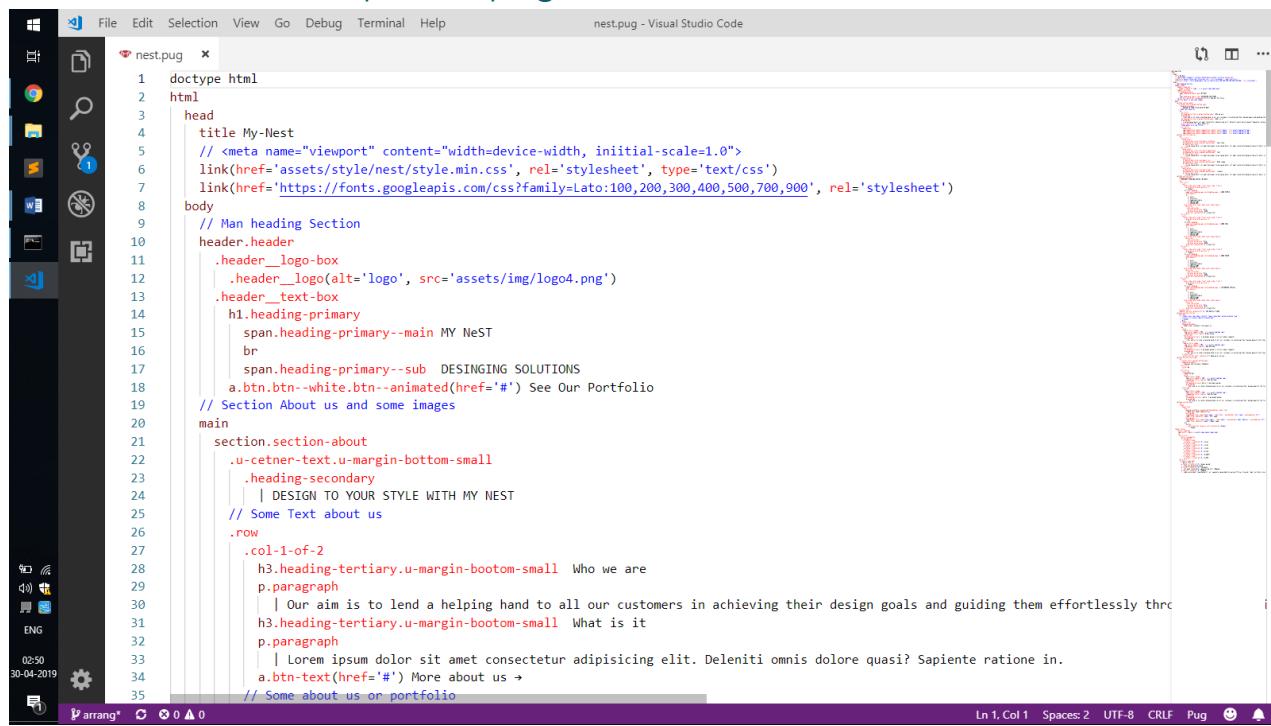
Pug is available via npm (GULP):

```
var pug = require('gulp-pug');

gulp.task('views', function buildHTML() {
  return gulp.src('views/*.pug')
    .pipe(pug({
      // Your options in here.
    }))
});
```

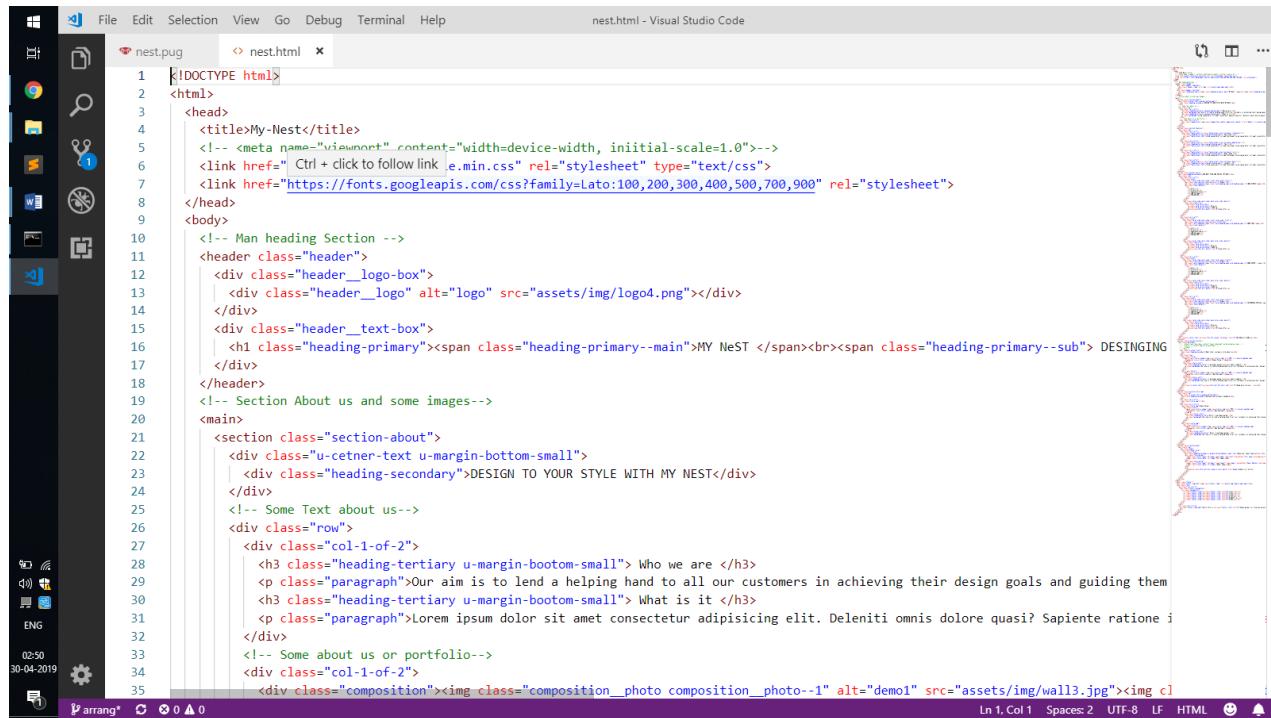
Let's see how pug code looks like.

PUG Structure | index.pug vs. index.html



The screenshot shows the Visual Studio Code interface with the file 'nest.pug' open. The code is a PUG template for a website. It includes a doctype declaration, an HTML tag, a head section with meta tags for viewport and links to CSS and Google Fonts. The body contains a header section with a logo and text, followed by a main section containing a heading and two paragraphs of descriptive text. The code uses indentation and comments to structure the template. The right side of the screen shows a detailed tree view of the project structure.

```
nest.pug
1 doctype html
2 html
3   head
4     title My-Nest
5     // <meta name="viewport" content="width=device-width, initial-scale=1.0">
6     link(href='assets/style/nest/style.min.css', rel='stylesheet', type='text/css')
7     link(href='https://fonts.googleapis.com/css?family=Lato:100,200,300,400,500,700,900', rel='stylesheet')
8   body
9     // Main heading Section
10    header.header
11      .header__logo-box
12        | .header__logo(alt='logo', src='assets/img/logo4.png')
13      .header__text-box
14        | h1.heading-primary
15          | span.heading-primary--main MY NeST
16        | br
17        | span.heading-primary--sub DESING SOLUTIONS
18        a.btn.btn--white.btn--animated(href='#') See Our Portfolio
19    // Section About us and some images
20    main
21      section.section-about
22        .u-cetner-text.u-margin-bottom-small
23        .heading-secondary
24          | DESIGN TO YOUR STYLE WITH MY NEST
25    // Some Text about us
26    .row
27      .col-1-of-2
28        h3.heading-tertiary.u-margin-bootom-small Who we are
29        p.paragraph
30          | Our aim is to lend a helping hand to all our customers in achieving their design goals and guiding them effortlessly thro
31        h3.heading-tertiary.u-margin-bootom-small What is it
32        p.paragraph
33          | Lorem ipsum dolor sit amet consectetur adipisicing elit. Deleniti omnis dolore quasi? Sapiente ratione in.
34        a.btn-text(href='#') More about us →
35    // Some about us or portfolio
```



The screenshot shows the Visual Studio Code interface with the file 'nest.html' open. This is the generated HTML output from the PUG template 'nest.pug'. The HTML structure is identical to the PUG template, including the doctype, HTML, head, and body sections. The right side of the screen shows a detailed tree view of the project structure.

```
nest.html
1 !DOCTYPE html
2 <html>
3   <head>
4     <title>My-Nest</title>
5     <!-- <meta name="viewport" content="width=device-width, initial-scale=1.0">-->
6     <link href="assets/style/nest/style.min.css" rel="stylesheet" type="text/css">
7     <link href="https://fonts.googleapis.com/css?family=Lato:100,200,300,400,500,700,900" rel="stylesheet">
8   </head>
9   <body>
10    <!-- Main heading Section -->
11    <header class="header">
12      <div class="header__logo-box">
13        | <div class="header__logo" alt="logo" src="assets/img/logo4.png"></div>
14      </div>
15      <div class="header__text-box">
16        | <h1 class="heading-primary"><span class="heading-primary--main">MY NeST </span><br><span class="heading-primary--sub">DESING
17        | </h1>
18      </div>
19    </header>
20    <!-- Section About us and some images-->
21    <main>
22      <section class="section-about">
23        <div class="u-cetner-text u-margin-bottom-small">
24          <div class="heading-secondary">DESIGN TO YOUR STYLE WITH MY NEST</div>
25        </div>
26        <!-- Some Text about us-->
27        <div class="row">
28          <div class="col-1-of-2">
29            <h3 class="heading-tertiary u-margin-bootom-small">Who we are </h3>
30            <p class="paragraph">Our aim is to lend a helping hand to all our customers in achieving their design goals and guiding them
31            <h3 class="heading-tertiary u-margin-bootom-small">What is it </h3>
32            <p class="paragraph">Lorem ipsum dolor sit amet consectetur adipisicing elit. Deleniti omnis dolore quasi? Sapiente ratione i
33          </div>
34        <!-- Some about us or portfolio-->
35        <div class="col-1-of-2">
36          <div class="composition"><img c
```

Frameworks & Libraries

Express.js

Express.js, or simply Express, is a web application framework for Node.js, released as free and open-source software under the MIT License. It is designed for building web applications and APIs. It has been called the de facto standard server framework for Node.js.

The original author, TJ Holowaychuk, described it as a Sinatra-inspired server[citation needed], meaning that it is relatively minimal with many features available as plugins. Express is the back-end component of the MEAN stack, together with the MongoDB database software and AngularJS front-end framework

TypeORM

TypeORM is an ORM that can run in NodeJS, Browser, Cordova, Phone Gap, Ionic, React Native, Native Script, Expo, and Electron platforms and can be used with Typescript and JavaScript (ES5, ES6, ES7, and ES8). Its goal is to always support the latest JavaScript features and provide additional features that help you to develop any kind of application that uses databases - from small applications with a few tables to large scale enterprise applications with multiple databases.

TypeORM supports both Active Record and Data Mapper patterns, unlike all other JavaScript ORMs currently in existence, which means you can write high quality, loosely coupled, and scalable, maintainable applications the most productive way.

TypeORM is highly influenced by other ORMs, such as Hibernate, Doctrine and Entity Framework.

JWT

JSON Web Token (JWT, sometimes pronounced) is an Internet standard for creating JSON-based access tokens that assert some number of claims. For example, a server could generate a token that has the claim "logged in as admin" and provide that to a client. The client could then use that token to prove that it is logged in as admin. The tokens are signed by one party's private key (usually the server's), so that both parties (the other already being, by some suitable and trustworthy means, in possession of the corresponding public key) are able to verify that the token is legitimate. The tokens are designed to be

compact, URL-safe, and usable especially in a web-browser single-sign-on (SSO) context. JWT claims can be typically used to pass identity of authenticated users between an identity provider and a service provider, or any other type of claims as required by business processes

FaceAPI

Face-api.js is a JavaScript API for face detection and face recognition in the browser implemented on top of the tensorflow.js core API. It implements a series of convolutional neural networks (CNNs), optimized for the web and for mobile devices.

TypeScript

TypeScript is an open-source programming language developed and maintained by Microsoft. It is a strict syntactical superset of JavaScript, and adds optional static typing to the language.

TypeScript is designed for development of large applications and trans compiles to JavaScript.^[5] As TypeScript is a superset of JavaScript, existing JavaScript programs are also valid TypeScript programs. TypeScript may be used to develop JavaScript applications for both client-side and server-side (Node.js) execution.

TypeScript originated from the shortcomings of JavaScript for the development of large-scale applications both at Microsoft and among their external customers. Challenges with dealing with complex JavaScript code led to demand for custom tooling to ease developing of components in the language

Date:: 18.July.2019

-----(Nodejs sequelize bulk insert) Customizing the Laravel View Path -

```
'paths' => [
    resource_path('views'),
    app_path('views'),
],
```

Src: <https://laravel-news.com/laravel-view-path>

Date:: 19.July.2019

-----(express get id in routes) Express router - :id?

This is an express middleware.

In this case, yes, it will route /todos/anything, and then req.params.id will be set to 'anything'

Src: <https://stackoverflow.com/questions/34095126/express-router-id>

Date:: 20.Jul.2019

(sequelize insert new record) insert a new record in nodejs using sequelize
POST methodSrc:

```
// parse application/x-www-form-urlencoded
app.use(bodyParser.urlencoded({ extended: false }))

// parse application/json
app.use(bodyParser.json())

app.post('/test', function (request, response) {
  return users.create({
    id: request.body.id,
    name: request.body.name,
    role: request.body.role,
    email: request.body.email
  }).then(function (users) {
    if (users) {
      response.send(users);
    } else {
      response.status(400).send('Error in insert new record');
    }
  });
});
```

Date:: 22.Jul.2019

(node js fetch json data to const) Reading JSON in Node.js

```
const http = require('http');

let req = http.get("http://localhost:8000/data.json", function(res) {
  let data = '',
    json_data;
  res.on('data', function(stream) {
    data += stream;
  });
  res.on('end', function() {
    json_data = JSON.parse(data);

    // will output a Javascript object
    console.log(json_data);
  });
});
req.on('error', function(e) {
  console.log(e.message);
});
```

Src: <https://usefulangle.com/post/109/nodejs-read-json>

Date:: 23.Jul.2019

-----(sequelizejs catch error) Where to handle the error in a sequelize ORM query statement?

```
models.Item.create({
  title : req.body.title,
  UserId : req.body.UserId
}).then(function(item){
  res.json({
    "Message" : "Created item.",
    "Item" : item
  });
}).catch(function (err) {
  // handle error;
});
```

Src: <https://stackoverflow.com/questions/35337738/where-to-handle-the-error-in-a-sequelize-orm-query-statement>

Date:: 24.Jul.2019

-----(pug loop Example) Pug loop – CodePen

```
-  
  
list = [{id:1, img:'', name:'dipen pamar'},{id:2, img:'',  
name:'Parmar'}]  
  
h1 Simple Loop Example (PUG)  
  
ul  
  
each val,i in list  
  
li=i + ", id: " + val.id + ", name: " + val.name
```

Src: <https://codepen.io/monochromer/pen/VWvvee>

Date:: 24.Jul.2019

-----(sequelize foreign key set) Sequelize.js foreign key

```
var Person = sequelize.define('Person', {
  name: Sequelize.STRING
});

var Father = sequelize.define('Father', {
  age: Sequelize.STRING,
  //The magic start here
  personId: {
    type: Sequelize.INTEGER,
    references: 'persons', // <<< Note, its table's name, not object name
    referencesKey: 'id' // <<< Note, its a column name
  }
});

PersonhasMany(Father); // Set one to many relationship
```

Src: <https://stackoverflow.com/questions/14169655/sequelize-js-foreign-key>

Date:: 25.Jul.2019

-----(html tr onclick checkbox js) click anywhere on a table row and it will check the checkbox its in...?

```
document.querySelector("table").addEventListener("click",
({target}) => {
  // discard direct clicks on input elements
  if (target.nodeName === "INPUT") return;
  // get the nearest tr
  if (tr) {
    // if it exists, get the first checkbox
    const checkbox =
      tr.querySelector("input[type='checkbox']");
  }
});
```

```
    if (checkbox) {  
        // if it exists, toggle the checked property  
        checkbox.checked = !checkbox.checked;  
    }  
}  
});
```

Src: <https://stackoverflow.com/questions/5833152/click-anywhere-on-a-table-row-and-it-will-check-the-checkbox-its-in>

_____ Date:: 26.Jul.2019 _____

----- (css select multiple attribute values) Specify multiple attribute selectors in CSS
Concatenate the attribute selectors:

```
input[name=Sex][value=M]
```

```
input[name="Sex"][value="M"]
```

Src: <https://stackoverflow.com/questions/12340737/specify-multiple-attribute-selectors-in-css>

_____ Date:: 26.Jul.2019 _____

----- (javascript check object attribute exists) How to check if object property exists with a variable holding the property name? [duplicate]

```
if('prop' in myObj){  
    alert("yes, i have that property");  
}
```

Src: <https://stackoverflow.com/questions/11040472/how-to-check-if-object-property-exists-with-a-variable-holding-the-property-name>

Date:: 27.Jul.2019

----- (sequelizejs store findAll data to let variable) Get only dataValues from Sequelize ORM

```
const users = await db.users.findAll({ })
    .map(el => el.get({ plain: true })) // add this line to code
```

Src: <https://stackoverflow.com/questions/46380563/get-only-datavalues-from-sequelize-orm>

----- (JS nested object to flat) Flatten javascript objects into a single-depth object

```
flattenObject: (ob) => {
  var toReturn = {};
  for (var i in ob) {
    if (!ob.hasOwnProperty(i)) continue;
    if ((typeof ob[i]) == 'object') {
      var flatObject = flattenObject(ob[i]);
      for (var x in flatObject) {
        if (!flatObject.hasOwnProperty(x)) continue;
        toReturn[i + '.' + x] = flatObject[x];
      }
    } else {
      toReturn[i] = ob[i];
    }
  }
  return toReturn;
}
```

Src: <https://gist.github.com/penguinboy/762197>

Date:: 31.Jul.2019

----- (get empty all values in html javascript) Clear all HTML fields using javascript

```
 $("input, input[type=text], textarea, input[type=email]").val('');
```

Src: <https://stackoverflow.com/questions/569357/clear-all-html-fields-using-javascript>

Date:: 1.Aug.2019

----- (target parent child of selected child) jQuery: How to get to a particular child of a parent?

```
$(".mylink").closest(".box").find("[class*='something']").first().attr("id")
$(this).closest('.box').children('.something1')
$(this).parent().siblings('.something1');
$(this).parents(".box").children(".something1");
$(this).parent().prev(); // if you always want the parent's previous sibling
```

Src: <https://stackoverflow.com/questions/2398947/jquery-how-to-get-to-a-particular-child-of-a-parent>

Date:: 20.Aug.2019

----- (typeorm export module connection manager) Complete guide to using TypeORM and TypeScript for data persistence in Node.js module

➔ OOPs with TypeORM Example Explained very well

Src: <https://levelup.gitconnected.com/complete-guide-to-using-typeorm-and-typescript-for-data-persistence-in-node-js-module-bfce169959d9>

Src: <https://techsparrx.com/nodejs/typescript/typeorm.html>

Date:: 22.Aug.2019

----- (Access current Request in Express Jade/ PUG) Google Title

var path = window.location.pathname;

```
$("a.nav-link[href='"+path+"']").parents('li.nav-item').addClass('active')
```

Src: /

Date:: 2.Oct.2019

-----(nodejs use Dirent ifFile) Node.js check if path is file or directory

`fs.lstatSync(path_string).isDirectory()` should tell you. From the [docs](#):

```
stats.isFile()  
stats.isDirectory()  
stats.isFIFO()  
stats.isSocket()
```

Src: <https://stackoverflow.com/questions/15630770/node-js-check-if-path-is-file-or-directory>